Information Systems Engineering and Management 61

S. D. Prabu Ragavendiran
Vasile Daniel Pavaloaia
M. S. Mekala
Selwyn Piramuthu *Editors*

Innovations and Advances in Cognitive Systems

ICIACS 2025, Volume 2

Springer

Information Systems Engineering and Management

Volume 61

The book series "Information Systems Engineering and Management" (ISEM) publishes innovative and original works in the various areas of planning, development, implementation, and management of information systems and technologies by enterprises, citizens, and society for the improvement of the socio-economic environment.

The series is multidisciplinary, focusing on technological, organizational, and social domains of information systems engineering and management. Manuscripts published in this book series focus on relevant problems and research in the planning, analysis, design, implementation, exploration, and management of all types of information systems and technologies. The series contains monographs, lecture notes, edited volumes, pedagogical and technical books as well as proceedings volumes.

Some topics/keywords to be considered in the ISEM book series are, but not limited to: Information Systems Planning; Information Systems Development; Exploration of Information Systems; Management of Information Systems; Blockchain Technology; Cloud Computing; Artificial Intelligence (AI) and Machine Learning; Big Data Analytics; Multimedia Systems; Computer Networks, Mobility and Pervasive Systems; IT Security, Ethics and Privacy; Cybersecurity; Digital Platforms and Services; Requirements Engineering; Software Engineering; Process and Knowledge Engineering; Security and Privacy Engineering, Autonomous Robotics; Human-Computer Interaction; Marketing and Information; Tourism and Information; Finance and Value; Decisions and Risk; Innovation and Projects; Strategy and People.

Indexed by Google Scholar. All books published in the series are submitted for consideration in the Web of Science.

For book or proceedings proposals please contact Alvaro Rocha (amrrocha@gmail.com).

S. D. Prabu Ragavendiran ·
Vasile Daniel Pavaloaia · M. S. Mekala ·
Selwyn Piramuthu
Editors

Innovations and Advances in Cognitive Systems

ICIACS 2025, Volume 2

Springer

Editors
S. D. Prabu Ragavendiran
Department of Computer Science
and Engineering
Builders Engineering College
Nathakadaiyur, Tamil Nadu, India

M. S. Mekala
Robert Gordon University
Aberdeen, UK

Vasile Daniel Pavaloaia
Artificial Intelligence
Alexandru Ioan Cuza University of Iasi
Iasi, Romania

Selwyn Piramuthu
Information Systems
University of Florida
Gainesville, FL, USA

ISSN 3004-958X ISSN 3004-9598 (electronic)
Information Systems Engineering and Management
ISBN 978-3-031-97712-1 ISBN 978-3-031-97713-8 (eBook)
https://doi.org/10.1007/978-3-031-97713-8

This volume is dedicated with sincere gratitude to all faculty members and the organizing committee members, whose expert guidance and commitment have been fundamental to the successful realization of this work. We also dedicate this book to the members of the Review Committee for their consistent support and efforts throughout the evaluation process. Finally, we extend our heartfelt thanks to all the contributing authors and participants, whose research and collaboration have brought this compilation to life.

Preface

We are pleased to present Volume 2 *Innovations and Advances in Cognitive Systems.* The volume focuses on smart technologies and their influence across various industries ranging from smart agriculture to medicine, from detecting financial fraud to accessibility in human-computer interaction. The chapters in this volume explores the widening scope of cognitive systems and the continuing thrust to impart intelligence into mundane environments.

This volume showcases a broad range of interdisciplinary studies, held together by a unifying theme: making intelligent systems smarter to sense, comprehend, and act on sophisticated real-world issues. Emphasizing application-focused innovation, the chapters illustrate how AI, machine learning, and deep learning methods are being aptly adapted to address problem-specific issues. Some of the highlights include cutting-edge research in smart agriculture as well as various deep learning methods and applications. Human-centric technology efforts are also well-represented through research on sign language interpretation, GAN-based emotion recognition, autonomous humanoid lab assistants, and federated learning models that maintain privacy while guaranteeing performance. This book also includes research studies on IoT-powered safety systems, smart fraud detection, and intelligent retail analytics. The wide range of subjects spanning handwritten text recognition and robotic automation, through communication-efficient learning frameworks and community-building platforms highlights the lively, interdisciplinary nature of cognitive systems research today.

We acknowledge the contributions of all the researchers, reviewers, and contributors to invaluable work on making this volume a reality. Through these insightful

studies and creative solutions, we hope to inspire new ideas, collaborations, and innovations in the dynamic domain of AI and cognitive computing.

Dr. S. D. Prabu Ragavendiran
Professor and Head
Department of Computer Science
and Engineering
Builders Engineering College
Nathakadaiyur, India

Dr. Vasile Daniel Pavaloaia
Professor of Artificial intelligence
Alexandru Ioan Cuza University of Iasi
Iasi, România

Dr. M. S. Mekala
Professor, Robert Gordon University
Aberdeen, UK

Dr. Selwyn Piramuthu
Professor of Information Systems
University of Florida
Gainesville, USA

Contents

Smart Agriculture: AI-Enabled Growth Prediction for Lettuce Cultivation

Prachi Sharma, Awanit Kumar, Nirmal Singh, Ajay Kumar Suwalka, Sheshang Degadwala, and Dhairya Vyas

Abstract Artificial intelligence in agriculture uses technology to improve crop growth and enhance predictions about harvests as well as resource distribution. The research uses multiple regression models to execute artificial intelligence-driven predictions of lettuce plant growth. The analysis of growth patterns used several data-driven approaches which included Linear Regression together with Decision Tree and K-Nearest Neighbors (KNN), Random Forest, and XGBoost methods. The Decision Tree model provided superior performance according to performance evaluation metrics where MSE scored 0.173, RMSE reached 0.417 and MAE amounted to 0.028, and R^2-score equaled 0.999. The results indicated that Linear Regression provided the minimum performance with 170.176 MSE and 0.005 R^2-score. XGBoost performed almost as well as Random Forest with a R^2-score of 0.931 while still showing strong accuracy rates. The study demonstrates that AI models have strong potential in precision agriculture through Decision Tree-based prediction which delivers improved results for predicting lettuce yield to enable advanced farming methods and sustainable food systems.

P. Sharma (✉) · A. Kumar · N. Singh · A. K. Suwalka
Sangam University, Bhilwara, Rajasthan, India
e-mail: ps411340@gmail.com

A. Kumar
e-mail: awanit.kumar@sangamuniversity.ac.in

N. Singh
e-mail: nirmal.singh@sangamuniversity.ac.in

A. K. Suwalka
e-mail: ajay.suwalka@sangamuniversity.ac.in

S. Degadwala
Department of Computer Engineering, Sigma University, Vadodara, Gujarat, India

D. Vyas
The Maharaja Sayajirao University of Baroda, Vadodara, Gujarat, India
e-mail: dhairya.vyas-cse@msubaroda.ac.in

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_1

Keywords Smart agriculture · AI prediction · Lettuce growth · Machine learning · Precision farming

1 Introduction

Traditional farming methods have evolved to modern technological practices through changes occurring in the agricultural sector. Modern agricultural practices have revolutionized crop cultivation and management through the implementation of artificial intelligence (AI) and machine learning (ML) and Internet of Things (IoT). AI technology started being used for agricultural applications during the early years of the twenty-first century through research investigations into data-based methods for increasing farming efficiency. The highly popular leafy vegetable known as lettuce needs exact environmental specifications to grow best. Basic farming operations mostly depend on human observation combined with experiential choices yet these techniques generate inconsistent crop yields and resource performance. AI-powered growth prediction models offer organizations the chance to maximize productivity by precisely forecasting crop development then intervening at strategic moments.

Precision agriculture advancements have not solved all difficulties in achieving accurate lettuce growth predictions. Traditional statistical methods together with simple regression approaches lack the capability to understand the complex yet non-linear connections between environmental elements and agricultural product yields. Studies indicate that Linear Regression and other conventional models demonstrate high error rates when processing datasets containing various features because these models do not handle large datasets effectively. Studies on predicting lettuce growth mainly utilize only one model while missing the evaluation between different machine learning techniques. A complete assessment of AI-based prediction models is necessary to establish the optimal method for lettuce cultivation because current gaps exist in the research field. Better agricultural performance and reduced waste together with enhanced resources distribution will result from fixing these operational flaws.

The main research objective explores how different machine learning algorithms function with AI to forecast lettuce growth through performance assessment. The research targets three main goals which encompass performance evaluation of regression models through standard metrics to establish a most accurate model and to demonstrate AI potential in smart agriculture optimization for lettuce production. Various regression models including Linear Regression, Decision Tree and K-Nearest Neighbors (KNN), Random Forest together with XGBoost underwent testing using appropriate datasets. The performance evaluation of four essential metrics including Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) together with Mean Absolute Error (MAE) and R^2-score was conducted. The selected metrics allow researchers to understand in detail how well each model predicts outcomes and maintains accuracy during the process.

The research results showed that Decision Tree model delivered the best results through its lowest error rates and highest 0.9989 R^2-score. The predictive capabilities of Linear Regression stood out as weak since its R^2-score approached zero. Random Forest and XGBoost models delivered precise results through their R^2 scores of 0.9405 and 0.9313 which indicates they are acceptable options for precision agriculture use. AI-driven methodologies used for predicting lettuce growth have established their effectiveness in modern farming by providing compelling results. The implementation of sophisticated AI models enables farmers to improve their yield predictions through better resource management systems thus supporting sustainable farming operations.

2 Literature Study

Kumaratenna et al. [1] created Planting-Density Growth Harvest (PGH) charts for controlled environment lettuce cultivation under artificial lighting conditions. Empirical modeling analysis of growth patterns occurred through the study under different planting density conditions. The research methodology worked to improve both lighting patterns and plant placement distances for achieving peak crop production efficiency. Strategic distribution of space between plants proved critical in leading to increased lettuce biomass production levels. The research project failed to include machine learning models as a method for generating automated yield forecasts. Prospective research should include artificial intelligence methods to optimize growth model frameworks which would enhance system scalability together with environmental adaptability. The research of Eshkabilov et al. [2] used hyperspectral imaging in combination with machine learning models to evaluate baby leaf lettuce nutrients. SVM, Random Forest and Partial Least Squares Regression (PLSR) operated together to forecast sugar, vitamin and nutrient values across hyperspectral spectrum information. The experimental outcomes showed that machine learning showed success in predicting nutrient levels and Random Forest delivered the most precise results. The research work did not include growth prediction as a part of its assessment process which limited its utility to measuring nutrients only. Future investigations should combine predictive systems for nutrients with growth modeling purposes for complete precision agriculture applications. Ojo et al. [3] conducted research to determine hydroponic lettuce phenotypic measures that improve resource utilization through their investigation. The research utilized image processing together with deep learning models especially Convolutional Neural Networks (CNNs) to obtain phenotypic traits from lettuce images. The deep learning algorithms enhanced leaf area and biomass prediction precision levels over conventional regression systems. The research stimuli did not include an assessment of different machine learning methods to provide comparative performance evaluation. Further research should develop hybrid artificial intelligence approaches which unite CNNs with ensemble learning methods to enhance plant growth predictions.

Hosoda et al. [4] created a fresh weight prediction model for plant factory lettuce using plant growth models. Nonlinear regression and mechanistic modeling served as the analytical tools for biomass simulation in this study. The results demonstrated that plant growth models estimated fresh weight successfully but operational success required predefined parameters. The absence of real-time machine learning in the study restricted its ability to adapt to different growing conditions. AI-driven real-time prediction models need implementation in future research to improve both the accuracy and adaptability of growth forecast models. The research by Rukumani Khandhan et al. [5] delivered lettuce growth modeling optimization by combining XGBoost with Support Vector Machines (SVM) supported by Gaussian Process Regression (GPR). The research evaluated both standalone performance and hybrid performance information from individual models as well as their resulting fusion-based hybrid framework. The hybrid model provided superior functionality when compared to sole operation of individual models as it produced more accurate predictions along with greater stability. The main focus of this research was algorithmic performance assessment instead of real-world implementation capability. Future investigations should focus on developing realistic methods for greenhouse implementation as well as real-time monitoring capabilities. Deep learning approaches investigated by Yu et al. [6] allowed researchers to both identify lettuce phenotypes precisely along with predicting their growth patterns. A CNN model was introduced by the study to process leaf morphology and growth patterns obtained from image datasets. The research method achieved accurate predictions of lettuce biomass and their phenotype variability. The research analyzed deep learning algorithms exclusively since it did not investigate traditional machine learning approaches for comparative results. Future research needs to investigate hybrid AI systems which combine with predictive accuracy monitoring through real-time analysis.

The authors Gowtham et al. [7] created an aeroponic lettuce crop growth monitoring system relying on machine learning algorithms. Decision trees and random forests operated in the research to forecast environmental elements alongside lettuce plant life stages. Prediction models based on machine learning improved accuracy levels when assessing plant growth performance over traditional methods. The research design had no adaptability in real-time operations because it failed to include implementation strategies for practical use. Future investigations need to combine monitoring systems built on IoT technology in order to enable real-time adjustments throughout aeroponic farming operations. The research by Gowtham et al. [8] utilized logistic regression to assess and predict lettuce production quantities in aeroponic vertical farming system. Researchers pursued supervised learning methods for the purpose of achieving maximum yield prediction performance. The results indicated that logistic regression showed moderate success in yield prediction for crops although it fell short compared to new generations of AI models. The researchers omitted deep learning and ensemble methods from their work which presents a future development opportunity. Future research needs to analyze how deep learning integration with logistic regression would improve prediction accuracy. The authors at Asy'ari et al. [9] used ARIMA autoregressive integrated moving average to forecast hydroponic lettuce farm growth. The research confirmed that using ARIMA allowed

successful predictions of future trends by analyzing historical patterns. The research did not provide evaluation results against contemporary artificial intelligence models which include gradient boosting and neural networks. Progressive research needs to implement artificial intelligence models with predictive functionalities to optimize hydroponic farming operations. Kallenberg et al. [10] performed research on combining process-based models with machine learning techniques for crop yield forecasting. The researchers improved yield prediction accuracy through an integration of experimental crop models and machine learning computation systems. The integrated model provided better performance than isolated crop models even though it needed substantial processing power. The research did not investigate immediate adjustments therefore lacking practical use cases. Research efforts should direct their efforts to developing enhanced hybrid models which can provide real-time yield forecasts.

The research of Rajendiran et al. [11] investigated how Random Forest Regression predicts yield data from aeroponic systems for lettuce crops. Research findings showed that random forest achieved superior performance when used to forecast lettuce yield through environmental parameter analysis. The research omitted investigation into deep learning algorithms and combination models that might enhance accuracy levels. The analysis of future studies requires combining ensemble learning systems with real-time data analytical methods for precision agriculture development. Machine learning regression combined with deep learning approaches served Sharma et al. [12] for agricultural yield prediction. Different regression models such as linear regression, decision trees and deep learning-based neural networks were analyzed against each other in yield estimation. Deep learning algorithms proved most effective for detecting intricate farming patterns based on their findings. The research examined other crops and omitted lettuce growth analysis precisely because it needed particular models tailored to individual agricultural products. Research in AI should concentrate on improving its effectiveness for lettuce cultivation needs. Kiremit et al. conducted research [13] to determine the optimal amounts of salicylic acid that would enhance lettuce resistance to salt stress. A controlled experimental design established by researchers determined the best salicylic acid concentration that led to maximized lettuce yield and stress resistance. Optimized dose treatments produced enhanced physiological characteristics as well as yield performance under salt stress conditions. The research failed to use predictive modeling as a method to determine dynamic optimal treatment numbers. Future work needs to merge AI-enabled methods which create real-time systems to help reduce stress.

Shalash et al. created machine learning models for hydroponic farming through their development of growth prediction and anomaly detection software [14]. A deep learning algorithm was used by the study to monitor plant health while forecasting future growth trends. The outcomes showed that machine learning models achieved success in detecting growth abnormality then generating improved cultivation plans. The research concentrated on a system driven by one model yet did not expand towards multiple model integration known as ensemble methods. Analysts need to develop combined artificial intelligence systems which enhance prediction stability as future research direction. Panigrahi et al. [15] performed a research on supervised

learning regression models to evaluate their effectiveness in crop yield prediction. This investigation evaluated linear regression together with decision trees and random forests and deep learning methods in their ability to forecast crop productivity. Deep learning models achieved the best accuracy rate in the experiments but these achievements were tied to expanding computational demands. The analysis omitted the investigation of live prediction features which made it ineffective for operational deployment. Research must concentrate on enhancing model performance so it can function effectively in smart agricultural real-time monitoring systems.

The current research on AI-enabled lettuce growth prediction shows particular weaknesses throughout the available studies. Previous scientific research either evaluated nutrient levels independently or estimated plant growth independently and does not combine both data sets for complete crop development analysis. The research includes several studies that continue using traditional modeling instead of adopting AI-based real-time prediction frameworks. Limited comparison research exists on the use of multiple AI models in spite of the incorporation of deep learning techniques by some studies. Future research needs to focus on the implementation of real-time artificial intelligence systems as well as combining statistical models with artificial intelligence and methodology for operational deployment to boost precision agriculture utilization.

3 Proposed Methodology

See Fig. 1.

3.1 Dataset

Recursive forecasting of lettuce growth periods requires systematic implementation of both data preparation and feature selection followed by model training and evaluation. A valuable plant growth dataset is available at Kaggle through the Lettuce Growth Days dataset which enables useful environmental condition insights. Five crucial attributes present in the dataset include temperature, humidity, pH level, Total Dissolved Solids (TDS) along with growth days to analyze development factors for plants. The structured dataset enables the creation of predictive models which forecast the full growth duration of lettuce plants. Research teams use these features to discover leading environmental factors which allow them to enhance agricultural practice optimization.

Link: https://www.kaggle.com/datasets/jjayfabor/lettuce-growth-days.

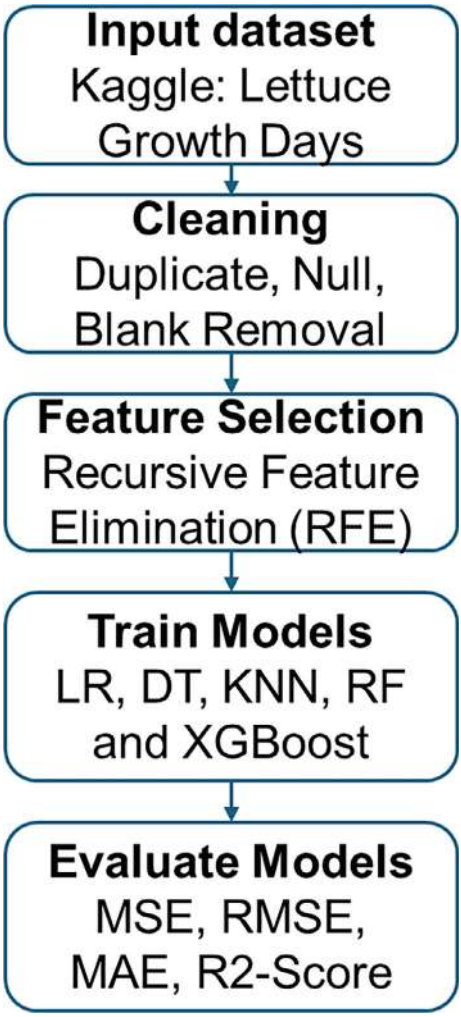


Fig. 1 Proposed work flow

3.2 Cleaning

Cleaning and preprocessing a dataset stands essential for training machine learning models because it guarantees both correctness and reliability. Raw datasets contain several types of inconsistencies such as empty fields and redundant records as well as absent values so these problems negatively affect model prediction accuracy. During the cleaning step duplicate entries receive removal as a protection against duplication and researchers decide between imputing relevant values or deleting those deemed insignificant. The data maintenance process includes a solution for dealing with empty spaces to protect dataset consistency. The quality of model predictions along

with performance increases when data cleaning operations improve input feature quality.

3.3 Feature Selection

In features selection operational phase after data cleaning entails the selection of vital variables that impact lettuce growth. The utilization of every available dataset feature at once can increase computational processes while potentially causing the model to overfit. The RFE technique serves as a solution to overcome these problems. RFE executes a process of model-based feature importance ranking to gradually remove unimportant traits. RFE enables a better model efficiency through variable selection because it allows meaningful predictions based on relevant data. Feature elimination stands as a vital stage because unneeded variables in the dataset introduce model distortions which degrade prediction accuracy. The process of feature selection works to lower the number of variables while simultaneously accelerating computations while enhancing the clarity of model interpretations.

3.4 Train Models

Various machine learning algorithms receive the selected relevant features for training different prediction models of lettuce plant growth duration. The selected algorithms undergo testing to find the most effective model among them. Linear Regression stands as the baseline model since it assumes environmental factors show a linear connection to lettuce growth. Complex interactions between various factors influence plant growth to such a degree that linear regression modeling might prove inadequate.

The non-linear relationships present a challenge which makes Decision Trees (DT) and Random Forest (RF) necessary models for analysis. Decision trees split datasets into branched segments according to feature values until they generate predictions at the concluding leaf points. Decision trees remain simple to use alongside being understandable but they tend to fit data points too closely when used for prediction. Random Forest through ensemble learning provides a solution to decision tree overfitting by generating numerous trees that produce their averaged predictions. The technique improves both model accuracy as well as reducing occurrences of overfitting.

K-Nearest Neighbors (KNN) serves as another machine learning tool for predicting growth days through data point similarity evaluation. The KNN prediction method calculates outcomes by aggregating the midpoint values of close neighboring points which deliver high effectiveness in patterned clustering data. Calculations in KNN models become inefficient as the size of available datasets grows larger.

The XGBoost algorithm serves as the last model because it provides superior gradient boosting functionality. The XGBoost algorithm operates as an ensemble method through sequential weak model generation so each new model enhances future predictions by rectifying previous errors. The model achieves high efficiency while delivering superior performance than other available algorithms for structured data structures so it works perfectly for plant growth prediction. The training process requires model input from processed data while engineers modify hyperparameters to enhance performance by using cross-validation procedures.

3.5 Evaluation of Model

The trained models need to undergo assessment for measuring their accuracy and predictive power regarding lettuce growth days determination. Model effectiveness depends on multiple assessment metrics that analysts use for evaluation purposes. The Mean Squared Error (MSE) serves as a primary evaluation metric to find the average squared differences between actual and predicted outcomes. The model provides more precise predictions when its MSE value decreases. The Root Mean Squared Error (RMSE) is used alongside MSE because MSE provides disproportionate weighting to large error values. To obtain a clear error magnitude measure RMSE extracts its square root value from MSE computations. The Mean Absolute Error (MAE) provides a calculation method for indicating the difference between actual values and predictions through absolute value comparison. The MAE calculation avoids square errors which means it exempts sensitivity to outliers. The method determines average absolute deviations between predicted and actual values without distortion through squaring. The R^2-score determines the amount of target variable variation which the model successfully predicts. The model demonstrates strong effectiveness in detecting environmental variations to plant growth days based on its R^2-score value.

4 Results Analysis

This research depicts the multiple phases of the machine learning pipeline which predicts lettuce growth through a collection of figures. Figure 2 displays the reading process of the dataset which shows that the information contains 3169 rows alongside 9 columns examining crucial environmental and growth factors. Figure 3 displays the correlation relationships between important variables which include Temperature (°C), Humidity (%), TDS Value (ppm) and pH Level and Growth Days (Fig. 4). Interface 4 demonstrates the positive relation between growth days and pH Level through its analysis. Figure 5 shows Recursive Feature Elimination demonstrated that Temperature (°C), TDS Value (ppm) alongside pH Level represent the essential factors affecting growth day predictions. Different regression models demonstrate

	Plant_ID	Date	Temperature (°C)	Humidity (%)	TDS Value (ppm)	pH Level	Growth Days	Temperature (F)	Humidity
0	1	8/3/2023	33.4	53	582	6.4	1	92.12	0.53
1	1	8/4/2023	33.5	53	451	6.1	2	92.30	0.53
2	1	8/5/2023	33.4	59	678	6.4	3	92.12	0.59
3	1	8/6/2023	33.4	68	420	6.4	4	92.12	0.68
4	1	8/7/2023	33.4	74	637	6.5	5	92.12	0.74
...	...	...	...	...	...	...	...	...	...
3164	70	9/13/2023	19.4	72	475	6.1	42	66.92	0.72
3165	70	9/14/2023	22.5	80	668	6.7	43	72.50	0.80
3166	70	9/15/2023	22.5	62	476	6.6	44	72.50	0.62
3167	70	9/16/2023	24.6	79	621	6.0	45	76.28	0.79
3168	70	9/17/2023	22.6	69	521	6.5	46	72.68	0.69

3169 rows × 9 columns

Fig. 2 Dataset reading

their performance according to the subsequent figures. The Linear Regression model shows the worst performance because it has an MSE value of 170.17 and an R^2-Score of 0.0053 demonstrating a very inadequate match (Fig. 6). The Decision Tree Regression model in Fig. 7 performs highly effectively with an MSE of 0.17 and an R^2-Score of 0.9989 to suggest nearly perfect prediction models. A moderate outcome emerges from K-Nearest Neighbors (KNN) Regression in Fig. 8 due to its MSE of 108.16 coupled with an R^2-Score of 0.3678. Random Forest Regression in Fig. 9 exhibits strong predictive ability through MSE = 10.17 with a R^2-Score of 0.9405. The performance of XGBoost Regression matches Random Forest because it shows MSE = 11.75 and R^2-Score = 0.9313 indicating ensemble approaches yield best prediction results (Fig. 10).

The performance statistics in Table 1 establish that the Decision Tree model delivers the best results by earning a 0.9989 R^2-Score and a low 0.1736 MSE. The Random Forest model along with XGBoost model produces reliable predictions through their R^2-Scores of 0.9405 and 0.9313 respectively. Linear Regression and KNN display poor performance in predicting lettuce growth since they produce higher MSE alongside reduced R^2-Scores which indicates their lack of effectiveness.

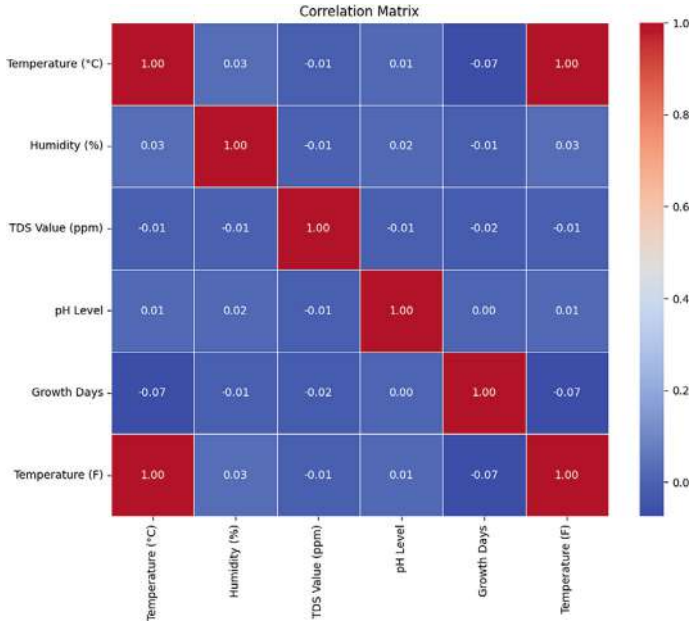


Fig. 3 Correlation map

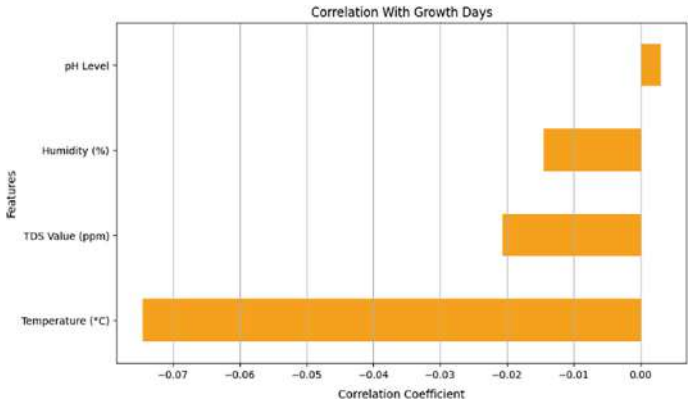


Fig. 4 Correlation with growth days

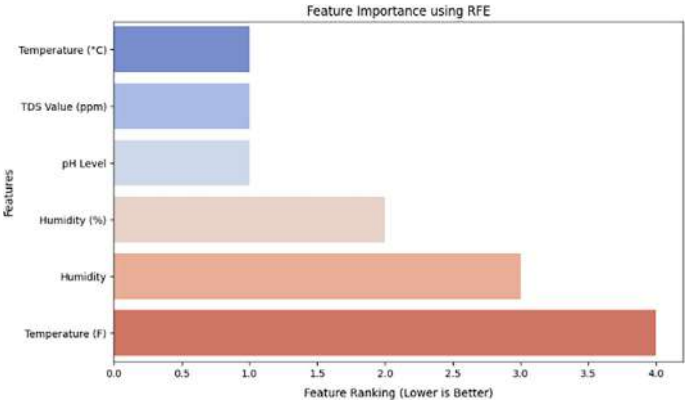
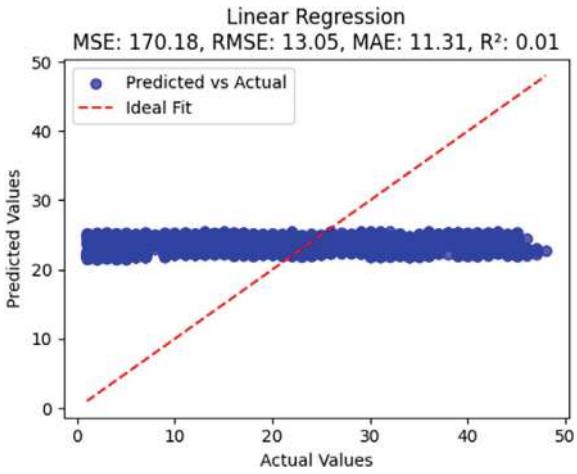


Fig. 5 Feature selection

Fig. 6 Linear regression



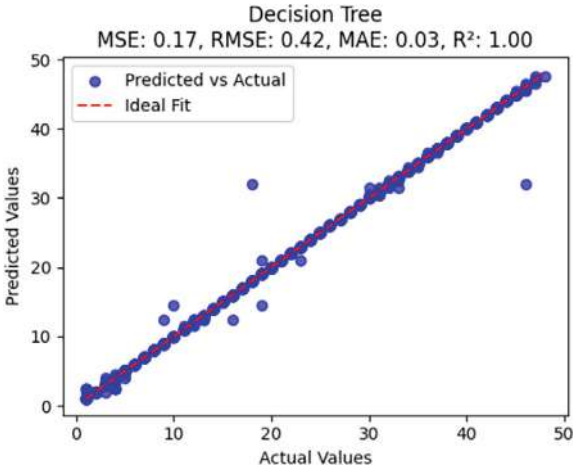


Fig. 7 Decision tree regression

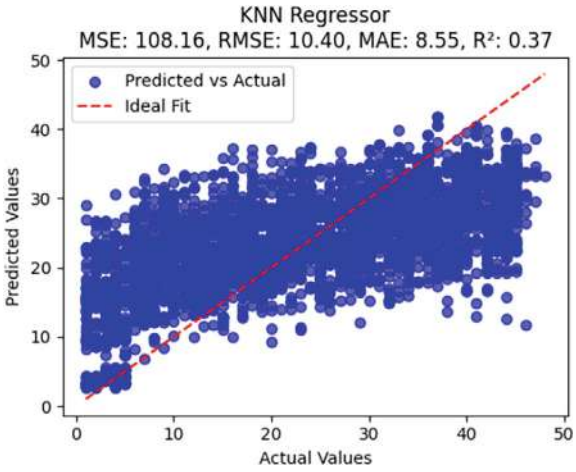


Fig. 8 K-nearest neighbour regression

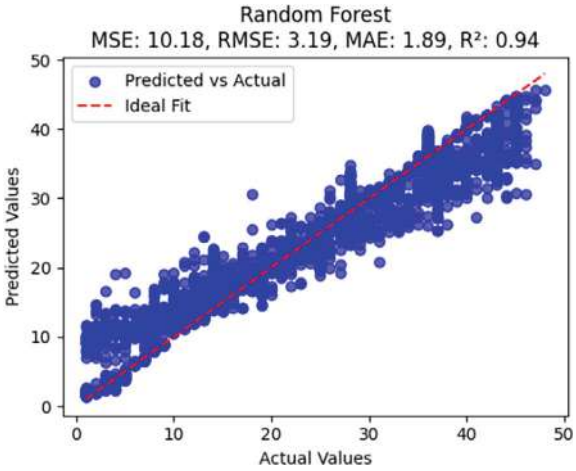


Fig. 9 Random forest regression

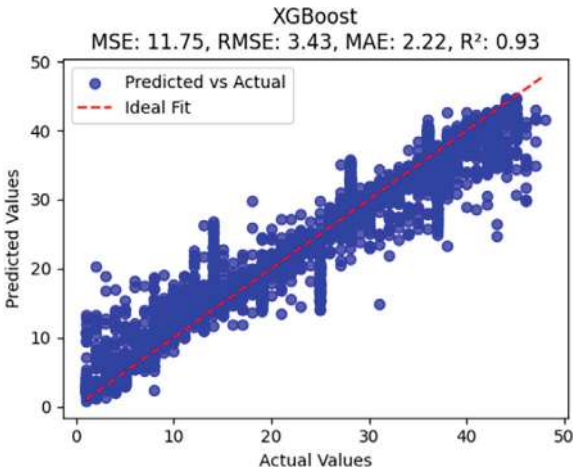


Fig. 10 XG-boost regression

Table 1 Evaluation of voting models

Model	MSE	RMSE	MAE	R^2-score
Linear regression	170.1762	13.0452	11.3147	0.0053
Decision tree	0.1736	0.4166	0.028	0.9989
KNN regressor	108.1555	10.3998	8.5458	0.3678
Random forest	10.1776	3.1902	1.8897	0.9405
XGBoost	11.7485	3.4276	2.2191	0.9313

5 Conclusion

The research has proven that machine learning algorithms precisely determine the length of lettuce growth by analyzing temperature alongside humidity and pH level along with TDS value. Decision Tree along with Random Forest and XGBoost outperform other regression models in this study because Decision Tree reaches an R^2-score of 0.9989 through data preprocessing and Recursive Feature Elimination (RFE) task and model evaluation processes. The study demonstrates how machine learning technology delivers accurate predictions to the field of agricultural planning. A deeper model improvement can happen by integrating diverse datasets and adding new environmental variables to build real-time monitoring systems. Future research should employ LSTMs alongside CNNs as deep learning methods to achieve superior modeling of complex non-linear dependencies. Implementation of IoT sensors for data collection in real-time along with automatic farmer recommendation systems will boost practical IoT applications. The approach would reach greater acceptance through extended testing across various crops grown in different climatic conditions. The implementation of AI predictive models in agriculture brings benefits to crop yield estimation while optimizing resources and promoting sustainable farming practices thus leading to data-driven and efficient agricultural practices.

References

1. Kumaratenna, K.P.S., Cho, Y.-Y.: Development of planting-density growth harvest (PGH) charts for lettuce grown in a plant factory with artificial lighting. Horticult. Environ. Biotechnol. 1–8 (2025)
2. Eshkabilov, S., Simko, I.: Assessing contents of sugars, vitamins, and nutrients in baby leaf lettuce from hyperspectral data with machine learning models. Agriculture (Switz.) **14**, 834 (2024). https://doi.org/10.3390/agriculture14060834
3. Ojo, M.O., Zahid, A., Masabni, J.G.: Estimating hydroponic lettuce phenotypic parameters for efficient resource allocation. Comput. Electron. Agric. **218**, 108642 (2024). https://doi.org/10.1016/j.compag.2024.108642
4. Hosoda, Y., Tada, T., Goto, H.: Lettuce fresh weight prediction in a plant factory using plant growth models. IEEE Access **12**, 97226–97234 (2024). https://doi.org/10.1109/ACCESS.2024.3423455
5. Rukumani Khandhan, C., Gothai, E., Kanagaraju, P., Rajkumar, S., Seenivasan, D., Anusurya, R.: Optimizing lettuce crop growth modeling with XGBoost-SVM and Gaussian process regression fusion. In: Lecture Notes in Networks and Systems, pp. 291–307. Springer (2024). https://doi.org/10.1007/978-981-97-7710-5_22
6. Yu, H., Dong, M., Zhao, R., Zhang, L., Sui, Y.: Research on precise phenotype identification and growth prediction of lettuce based on deep learning. Environ. Res. **252**, 118845 (2024)
7. Gowtham, R., Jebakumar, R.: A machine learning approach for aeroponic lettuce crop growth monitoring system. In: Lecture Notes in Networks and Systems, pp. 99–116. Springer (2023). https://doi.org/10.1007/978-981-99-1726-6_9
8. Gowtham, R., Jebakumar, R.: Analysis and prediction of lettuce crop yield in aeroponic vertical farming using logistic regression method. In: 2nd International Conference on Sustainable Computing and Data Communication Systems, ICSCDS 2023—Proceedings, pp. 759–764. IEEE (2023). https://doi.org/10.1109/ICSCDS56580.2023.10104763.

9. Asy'ari, M.Z., Aten, J.F.C., Prasetyo, D.: Growth predictions of lettuce in hydroponic farm using autoregressive integrated moving average model. Bull. Electr. Eng. Inf. **12**, 3562–3570 (2023). https://doi.org/10.11591/eei.v12i6.4820
10. Kallenberg, M.G.J., Maestrini, B., van Bree, R., Ravensbergen, P., Pylianidis, C., van Evert, F., Athanasiadis, I.N.: Integrating processed-based models and machine learning for crop yield prediction. arXiv preprint arXiv:2307.13466 (2023)
11. Rajendiran, G., Rethnaraj, J.: Lettuce crop yield prediction analysis using random forest regression machine learning model in aeroponics system. In: Proceedings of the 2023 2nd International Conference on Augmented Intelligence and Sustainable Systems, ICAISS 2023, pp. 565–572. IEEE (2023). https://doi.org/10.1109/ICAISS58487.2023.10250535
12. Sharma, P., Dadheech, P., Aneja, N., Aneja, S.: Predicting agriculture yields based on machine learning using regression and deep learning. IEEE Access **11**, 111255–111264 (2023). https://doi.org/10.1109/ACCESS.2023.3321861
13. Kiremit, M.S.: Erratum to: optimization of salicylic acid dose to improve lettuce growth, physiology and yield under salt stress conditions (Gesunde Pflanzen, (2023). https://doi.org/10.1007/s10343-023-00930-4). Gesunde Pflanzen **76**, 269–283 (2023). https://doi.org/10.1007/s10343-023-00945-x
14. Shalash, O., Métwalli, A., Elhefny, A., Rezk, N., El Gohary, F., El Hennawy, O., Akrab, F., Shawky, A., Mohamed, Z., Hassan, N.: Enhancing Hydroponic Farming with Machine Learning: Growth Prediction and Anomaly Detection. Available at SSRN 5079228 (2023)
15. Panigrahi, B., Kathala, K.C.R., Sujatha, M.: A machine learning-based comparative approach to predict the crop yield using supervised learning with regression models. Proc. Comput. Sci. **218**, 2684–2693 (2023). https://doi.org/10.1016/j.procs.2023.01.241

Neural Network-Based Smart Detection of Skin Cancer Using Radial Basis Function Networks

S. D. Vijayakumar, G. Vijayakumari, R. Praveenkumar, V. Kumar, T. Velmurugan, and G. Brinda

Abstract Skin cancer stands as the most prevalent cancer type in the United States, with an annual diagnosis of over 5 million cases. The timely identification and treatment of skin cancer play a pivotal role in enhancing patient prognosis. Machine learning has emerged as a promising avenue for aiding in the early detection of skin cancer, and the Radial Basis Function (RBF) approach has gained popularity as a technique in this regard. RBF networks, a subtype of artificial neural networks, utilize radial basis functions as activation functions. These functions, represented by bell-shaped curves, yield output values based on the distance between the input and the function's center. RBF networks have demonstrated effectiveness in classifying intricate data, making them well-suited for the detection of skin cancer. Among skin cancers, melanoma, originating from melanocytes—the pigment-producing cells—is the most perilous form and has been increasingly identified as a leading cause of death. Melanoma presents itself with regions appearing black or brown due to the melanin pigment. However, some melanomas do not produce melanin, manifesting in pink, tan, or white colors. Therefore, an efficient melanoma detection technique becomes imperative. RBFN, falling under the category of Artificial Neural Networks (ANN), has found utility in various classification problems in science and engineering. The Back Propagation (BP) algorithm, widely used in ANN, suffers from

G. Vijayakumari
Department of Electronics and Communication Engineering, Builders Engineering College, Tirupur, India

G. Brinda
Department of Electronics and Communication Engineering, M. P. Nachi Muthu M. Jaganathan Engineering College, Erode, India

S. D. Vijayakumar (✉) · R. Praveenkumar
Department of Computer Science Engineering, Nandha Engineering College, Erode, India
e-mail: mail2vijay.sd@gmail.com

T. Velmurugan
Department of Computer Science and Design, Kongu Engineering College, Perundurai, India

V. Kumar
Department of Electronics and Communication Engineering, Hindustan College of Engineering, Perundurai, India

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_2

drawbacks such as slow error rate convergence and susceptibility to getting stuck at local minima. To address these issues, a recent MATLAB tool has been employed for implementing the proposed system, designed using High-Level Synthesis (HLS) design methodology.

Keywords Melanoma · Radial basis function · Artificial neural networks. MATLAB · High-level synthesis

1 Introduction

Skin cancers originate from the skin due to the development of abnormal cells capable of invading or spreading to other body parts. They encompass three primary types: basal-cell skin cancer (BCC), squamous cell skin cancer (SCC), and melanoma. Basal-cell and squamous-cell cancers, along with less frequent types, constitute non-melanoma skin cancer (NMSC). Basal-cell cancer typically progresses slowly, potentially causing local tissue damage but generally not spreading distantly or proving fatal. It commonly manifests as a painless raised area of skin, sometimes shiny with visible blood vessels, or as a raised area with an ulcer. Squamous-cell skin cancer is more prone to spreading and typically presents as a hard lump with a scaly top or an ulcer. Melanomas are the most aggressive form, characterized by changes in size, shape, color, irregular edges, itchiness, or bleeding in moles. Skin lacking adequate melanin is susceptible to sunburn and harmful UV rays. Clinical examination and biopsy are standard diagnostic procedures, with dermatologists employing dermatoscopy—a magnifying optical device—to analyze skin structure. Early diagnosis of melanoma is crucial, as research indicates it's easier to control or prevent in its initial stages. Various diagnostic methods exist for melanoma, including the Seven Point Checklist, CASH (Color, Architecture, Symmetry, and Homogeneity) criteria. Melanoma rates have been steadily rising for three decades, with white individuals being 20 times more susceptible compared to African Americans. The lifetime risk of melanoma is approximately 2% for whites, 0.1% for blacks, and 0.5% for Hispanics.

In the context of standard regression for function approximation, we operate with a dataset comprising N training points within a D-dimensional input space. Each input vector $x(p) = \{x_i^p: i = 1, \ldots, D\}$ corresponds to a K-dimensional target output $t(p) = \{t_k^p: k = 1, \ldots, K\}$. Typically, these target outputs stem from underlying functions g (k(x)) combined with random noise, where the functions have centers $\{\mu_j\}$ and widths $\{\sigma_j\}$. The logical progression involves devising a method to determine suitable values for M, $\{w_{kj}\}$, $\{\mu_{ij}\}$ and $\{\sigma_j\}$.

The RBF Mapping can be framed akin to a neural network: the hidden-to-output layer behaves akin to a conventional feed-forward MLP network, where the summation of weighted hidden unit activations yields output unit activations. Hidden unit activations stem from basis functions $\varphi_j(x, \mu_j, \sigma_j)$, contingent upon "weights"$\{\mu_{ij}, \sigma_j\}$ and input activations $\{x_i\}$ in a non-standard manner.

RBFNs are a type of artificial neural network that utilizes radial basis functions as activation functions to efficiently classify data with non-linear patterns. In this system, preprocessing techniques such as median filtering are applied to enhance image quality, while feature extraction focuses on detecting critical attributes like texture, color, and shape. The classification process in RBFNs provides fast learning, smooth decision boundaries, and high accuracy, making them ideal for melanoma detection. This approach enhances early diagnosis, reduces manual workload, and improves decision-making, ultimately contributing to better patient outcomes and a higher survival rate. Skin cancer, especially melanoma, is a growing global health concern. Traditional diagnostic methods rely on visual inspection and biopsy, which can be time-consuming and subjective. RBFN-based smart detection systems address these limitations by automating the classification of skin lesions, significantly improving diagnostic accuracy. The process includes preprocessing (median filtering), feature extraction (color, texture, shape), and classification using RBFN. With its high accuracy, smooth decision boundaries, and efficient learning, RBFN helps in early melanoma detection, reducing misdiagnosis risks and enabling timely medical intervention. By integrating AI-driven diagnostics, RBFN enhances clinical decision-making, ultimately leading to improved patient survival rates and better healthcare outcomes.

Skin cancer, especially melanoma, is a growing global health concern. Traditional diagnostic methods rely on visual inspection and biopsy, which can be time-consuming and subjective. RBFN-based smart detection systems address these limitations by automating the classification of skin lesions, significantly improving diagnostic accuracy. The process includes preprocessing (median filtering), feature extraction (color, texture, shape), and classification using RBFN. With its high accuracy, smooth decision boundaries, and efficient learning, RBFN helps in early melanoma detection, reducing misdiagnosis risks and enabling timely medical intervention. By integrating AI-driven diagnostics, RBFN enhances clinical decision-making, ultimately leading to improved patient survival rates and better healthcare outcomes.

Reliability is achieved through:

- **Preprocessing techniques** like median filtering and histogram equalization to ensure high-quality inputs.
- **Robust feature extraction** methods focusing on color, texture (GLCM), and shape features (ABCD criteria).
- Use of **Radial Basis Function Networks (RBFNs)**, which offer **smooth decision boundaries** and **low Mean Squared Error (MSE)**, resulting in consistent and dependable outputs.
- **Validation with multiple images** (normal vs affected) and achieving classification accuracy over **99%**, demonstrating model robustness.

2 Literature Review

Skin cancer, particularly melanoma, is the deadliest form of cancer because it is highly fatal if not diagnosed in time. With the recent progress in artificial intelligence (AI) and deep learning, AI-based computer-aided systems for the diagnosis of skin cancer have attracted considerable attention in recent years. This review discusses various methodologies which have been suggested by researchers to identify and classify skin cancer using AI-based systems. Various researchers have come up with non-invasive automatic skin lesion analysis techniques in real-time. Abuzaghleh et al. [1] have come up with a system for early detection of melanoma based on real-time image processing methodologies. Their system combines feature extraction and classification mechanisms for the efficient detection of melanoma. Segmentation is an important process in computer-aided diagnosis systems. Santy and Joseph [2] discussed different segmentation techniques, with emphasis on thresholding, edge detection, and clustering algorithms for melanoma lesion segmentation. Likewise, Joseph and Panicker [3] discussed a fast hair segmentation technique that improves the rate of classification of skin lesions by removing noise from dermoscopic images.

Application of machine learning (ML) and deep learning (DL) techniques has significantly progressed the area of skin cancer classification. Burada et al. [4] suggested a computer-aided diagnosis system from radial basis function networks, which demonstrated a good success rate for melanoma classification. Balaji et al. [5] compared several neural network architectures and utilized the Firefly optimization algorithm to maximize the accuracy and precision of skin cancer classification. Likewise, Lakshmi and Jasmine [6] suggested a hybrid artificial intelligence model using a variety of ML approaches to improve diagnostic performance.

Nyemeesha and Ismail [7] conducted a systematic review of the employment of back-propagated artificial neural networks to detect skin cancer and their ability to identify multiple types of skin lesions. Xu et al. [8] used soft computing techniques for computer-aided diagnosis and discussed feature extraction and classification techniques. Kumar et al. [9] discussed the evolving landscape of artificial intelligence in detecting melanoma and analyzed various deep learning architectures and their impact on healthcare. Deep learning algorithms, especially convolutional neural networks (CNNs), have been extremely accurate in the diagnosis of skin cancer. SkNet, a CNN-based classifier, was presented by Jeny et al. [10] that properly distinguishes among different classes of skin cancer. Dildar et al. [11] discussed deep learning methods in the diagnosis of skin cancer by comparing various architectures and their accuracy in diagnosis.

There have been various studies providing comprehensive reviews of ML and DL models employed to predict diseases. Roobini et al. [12] provided a comprehensive review of ML-based models for disease prediction, emphasizing their significance in skin cancer diagnosis. Karpakam et al. [13] ventured into the analysis of fuzzy decision support systems and deep learning in aiding visually impaired people and establishing the prospects of AI application around medical diagnosis. Chowdary et al. [14] conceptualized an insulin dose predicting smart system, highlighting the

potential of DL and ML application in personal health solutions. The presented work demonstrates remarkable advances in applying AI for skin cancer identification. A variety of methods have been applied, ranging from classical segmentation algorithms to contemporary deep learning algorithms, to enhance melanoma detection speed and accuracy. Future studies must be aimed at integrating various data types, analyzing the results, and applying AI diagnosis systems to clinical practice.

2.1 Training RBF Networks

While the computational power proofs delineate the capabilities of RBF Networks, they offer no insights into parameter/weight optimization $\{M, w_{kj}, \mu_j \sigma\}., \sigma\}$. Unlike MLPs, RBF network layers operate differently, warranting distinct learning algorithms. Input-to-hidden "weights" $\{\mu_{ij}, \sigma_j\}$ can be trained via various unsupervised techniques, while the subsequent training focuses on hidden-to-output weights $\{w_{kj}\}$ using simple matrix pseudo-inversion.

2.2 Basis Function Optimization

Radial Basis Function Networks (RBFNs) rely on basis function optimization to enhance classification accuracy and generalization performance. Unlike traditional fully connected neural networks that require complex backpropagation-based training, RBFNs optimize basis function parameters efficiently without full nonlinear optimization. The effectiveness of RBFNs is largely dependent on how well the centers, widths (spread), and number of basis functions (M) are chosen. Several methods exist to determine these parameters, with three primary approaches being random selection, clustering-based approaches, and orthogonal least squares (OLS) regression.

2.2.1 Fixed Centers Selected at Random

One of the simplest ways to optimize basis functions in RBFNs is to randomly select centers from the training dataset. This method assumes that randomly chosen centers can sufficiently represent the underlying data distribution.

2.2.2 Clustering-Based Approaches (K-Means, Fuzzy C-Means, Etc.)

Clustering methods like K-Means or Fuzzy C-Means (FCM) are often used to determine optimal center locations in RBFNs. These techniques partition the input data into M clusters, with each cluster center representing a radial basis function center.

2.2.3 Orthogonal Least Squares (OLS) Regression

Orthogonal Least Squares (OLS) is a supervised learning approach that optimally selects basis function centers by minimizing the reconstruction error. Unlike unsupervised methods, OLS iteratively selects the most significant basis function centers while removing redundant ones.

2.3 Fixed Centers Selected at Random

The simplest approach for optimizing Radial Basis Function Networks (RBFNs) involves fixing the centers of basis functions at M randomly selected points from a given dataset containing N data points. This method assumes that randomly chosen centers can adequately represent the underlying data distribution. Once the centers are selected, the widths (σ_j) of the radial basis functions are set equally across all basis functions, ensuring uniform representation of data clusters.

The choice of σ_j is typically based on the maximum or average distance between the selected centers, a common formula being $\sigma_j = \text{dmax}/(2\text{M})$, where dmax represents the maximum pairwise distance between selected centers, and M is the number of basis functions. By maintaining suitable widths relative to the data distribution, individual RBFs can capture local variations effectively without excessive overlap or excessive separation. This ensures that each basis function contributes meaningfully to the classification process, maintaining an appropriate degree of generalization.

The random selection of centers, combined with a uniform spread parameter, works effectively for large datasets, where sufficient coverage of the feature space is likely. However, in smaller or highly imbalanced datasets, this method may lead to suboptimal center placement, reducing classification accuracy. Despite its simplicity, this approach is widely used due to its low computational cost and ease of implementation. It provides a reasonable approximation for basis function optimization, especially when combined with validation-based tuning of the number of basis functions (M) to prevent underfitting or overfitting.

3 Proposed Methodology

3.1 Image Acquisition

Image acquisition is the first step in detecting skin cancer, where high-resolution images of the skin are captured using digital cameras, dermatoscopes, or medical imaging sensors. These images serve as input for further processing. Proper lighting and high-quality imaging are essential to ensure accurate analysis. The collected

images are then stored in standardized formats, such as JPEG, PNG, or DICOM, for consistency in analysis.

3.2 Image Preprocessing

Preprocessing enhances the quality of the acquired image by removing unwanted noise and improving contrast. Techniques like Gaussian filters help in noise reduction, while Histogram Equalization enhances the visibility of skin lesions. Segmentation methods such as Thresholding or Edge Detection help in isolating the affected skin region from the background. Normalization is also performed to standardize the size and intensity of images before further analysis.

3.3 Feature Extraction

Feature extraction involves identifying key characteristics from the image that help differentiate normal and cancerous skin. These features include texture, color, and shape. Texture features (e.g., Gray Level Co-occurrence Matrix—GLCM) analyze surface patterns. Color features (e.g., RGB and HSV values) help detect pigmentation irregularities. Shape features (e.g., Asymmetry, Border Irregularity, and Diameter) follow the ABCD rule, commonly used in melanoma detection.

3.4 Classification Using Radial Basis Function Networks (RBFN)

Radial Basis Function Networks (RBFN) are a type of artificial neural network (ANN) commonly used for classification problems, including skin cancer detection. Artificial Neural Networks (ANNs), particularly **Radial Basis Function Networks**, serve as:

- **Core classifiers** in the system, replacing subjective manual inspection.
- Tools that map non-linear feature spaces effectively using **Gaussian activation functions**.
- Fast learners with **low training complexity**, making them ideal for **real-time medical diagnostics**.
- ANN here is used not as a black-box model but as a **three-layer architecture** (input-hidden-output), designed to capture complex skin lesion patterns.

RBFNs consist of three main layers: the input layer, hidden layer, and output layer. The input layer receives extracted features such as texture, color, and shape from the

preprocessed skin image. The hidden layer applies radial basis functions, typically Gaussian functions, to compute the similarity between input features and known patterns. The number of neurons in this layer determines the model's accuracy, with a higher number improving classification up to a certain point. The output layer classifies the lesion as benign, malignant, or normal based on probability scores. RBFNs are preferred for medical image classification because they provide fast learning, smooth decision boundaries, and high accuracy. By leveraging feature-based learning, RBFNs enable reliable and early skin cancer detection, assisting in automated diagnosis and medical decision-making.

4 Results and Discussion

The comparison between the non-affected and affected skin images demonstrates the effectiveness of the melanoma disease analyzer in identifying abnormalities based on image processing techniques, feature extraction, and classification models. The non-affected image exhibits a smooth and uniform texture in the RGB, black-and-white (BW), and semi-trace representations, with minimal intensity variations and dominant green bars in the density and trace occurrence data plots, indicating normal skin conditions.

In contrast, the affected image reveals irregular pigmentation, high-density traces, and concentrated white pixel clusters in the BW representation, signaling abnormal skin structure. The RGB representation in the affected image shows a circular pattern with increased blue and yellow intensities, while the semi-trace image displays denser red traces, highlighting potential melanoma-affected regions. Furthermore, the density and trace occurrence plots of the affected image exhibit higher red and pink bars, confirming a high percentage of affected pixel points. These differences validate the system's ability to analyze variations in skin texture, color, and intensity, thereby improving the early detection of melanoma. By leveraging advanced image processing and neural network-based classification, this system offers a reliable, automated approach for diagnosing skin cancer, minimizing misdiagnosis risks, and enhancing clinical decision-making.

Classification is achieved through:

- **Input Layer**: Receives extracted features (texture, color, shape).
- **Hidden Layer**: Applies Gaussian radial basis functions to measure feature similarity.
- **Output Layer**: Uses weighted summation of hidden layer outputs to produce classification results (e.g., benign, malignant).
- **Training Steps**:
  - Feature vectors labeled and fed to the RBFN
  - Hidden layer centers optimized via clustering or OLS
  - Output layer weights calculated via pseudo-inverse or gradient descent.

- **Result**: Skin lesion is classified with high accuracy and precision, aiding early diagnosis.

4.1 Normal Image

The initial input consists of a non-affected image of the skin. Subsequently, it undergoes several processes including pre-processing, feature extraction, and feature classification. The resulting output is illustrated below in Fig. 1.

4.1.1 Command Window for Normal Image

Within this command window, pixel values, trace values, and the disease stage are presented. Additionally, it show cases the occurrence percentage of the disease, as depicted in Fig. 2.

Here the occurrence percentages (both density-based and trace-based) are 0, the system confirms that the analyzed skin image is normal (non-affected) with no signs of melanoma. The neural network-based classification model demonstrates high accuracy, consistently achieving over 99%, particularly when using an optimal number of neurons. While increasing the number of neurons slightly improves the Mean Squared Error (MSE) and classification accuracy, the improvements plateau beyond 50 neurons, indicating a point of diminishing returns. This suggests that adding more neurons does not significantly enhance performance and may only increase computational complexity. Overall, the classification system effectively identifies non-affected skin, proving to be a reliable tool for early melanoma detection. Since the input image in this case is non-affected, the system likely identifies minimal or no traces of melanoma, reinforcing its capability to distinguish between normal and affected skin conditions. This demonstrates the effectiveness of

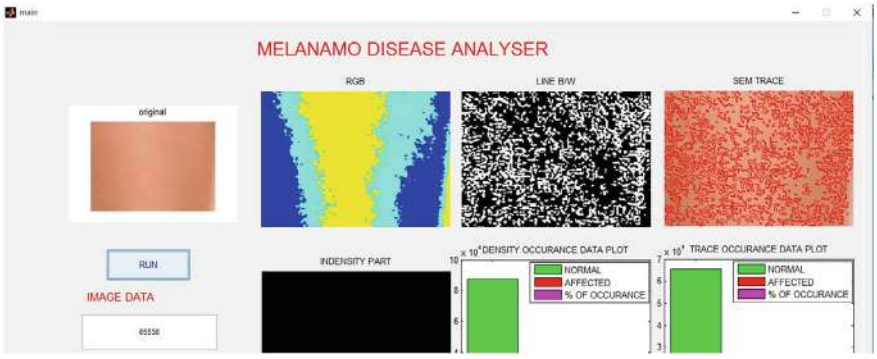


Fig. 1 Non-affected image output

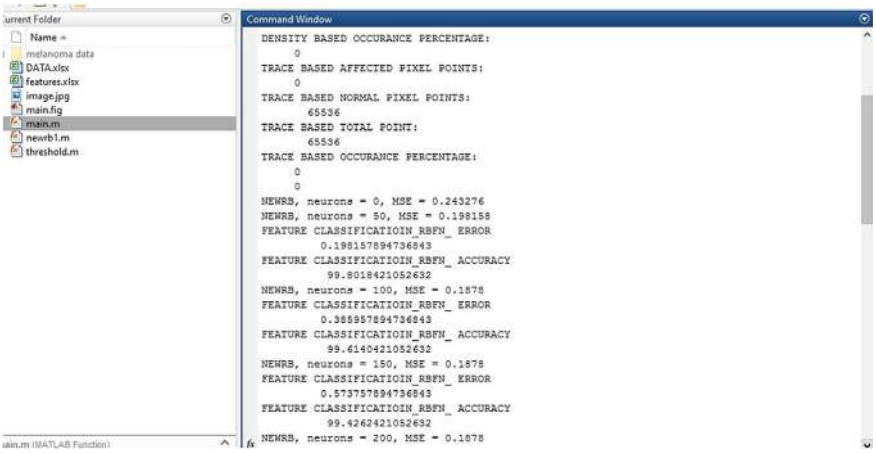


Fig. 2 Command window for non-affected image

the analyzer in early detection and classification, which can aid in timely medical intervention.

4.1.2 Performance Graph for Normal Image

This graph illustrates the progression of the disease stages. It depicts the performance of the non-affected image, as shown in Fig. 3. It can be generated using **MATLAB's Neural Network Toolbox**, by training the model on non-affected skin images.

4.2 Affected Image

The input initially consists of the affected image of the skin. Subsequently, it undergoes several processes including pre-processing, feature extraction, and feature classification. The resulting output is depicted in Fig. 4.

4.2.1 Command Window for Affected Image

In this command window, the pixel value and trace value of the affected image are displayed, along with the stage of the disease occurrence. Additionally, the window presents the percentage of disease occurrence, as illustrated in Fig. 5.

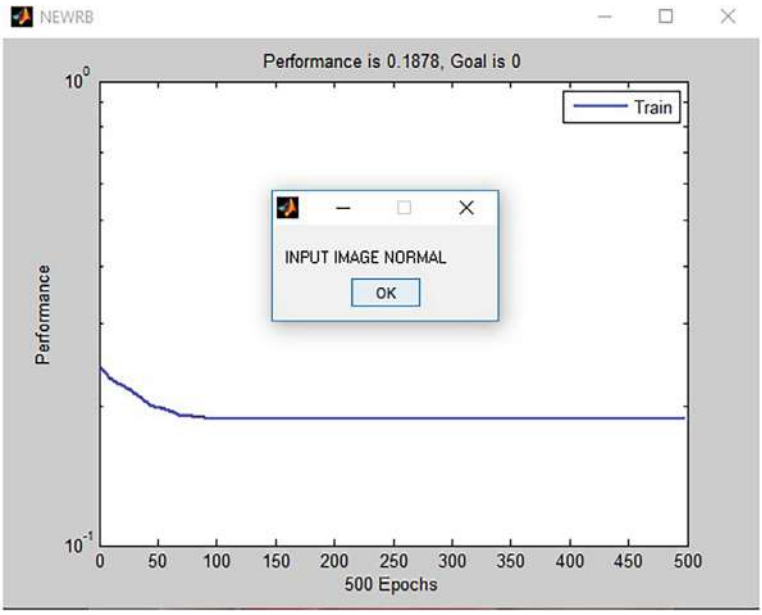


Fig. 3 Non-affected image graph

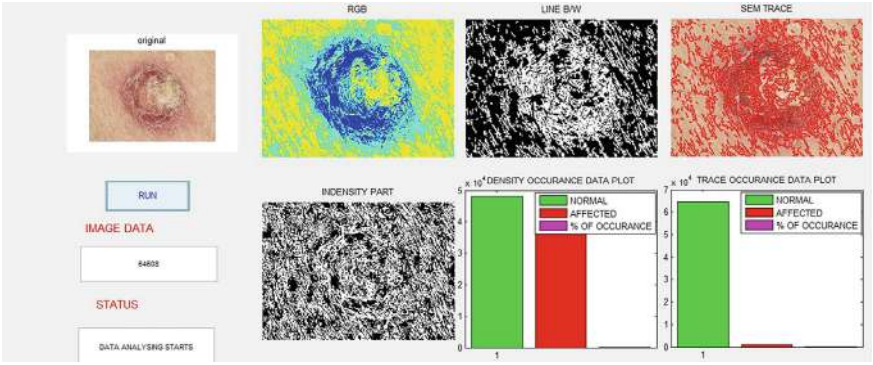


Fig. 4 Affected image output

4.2.2 Performance Graph of Affected Image

This graph illustrates the progression of the disease stages. It represents the performance of the non-affected image, as depicted in Fig. 6.

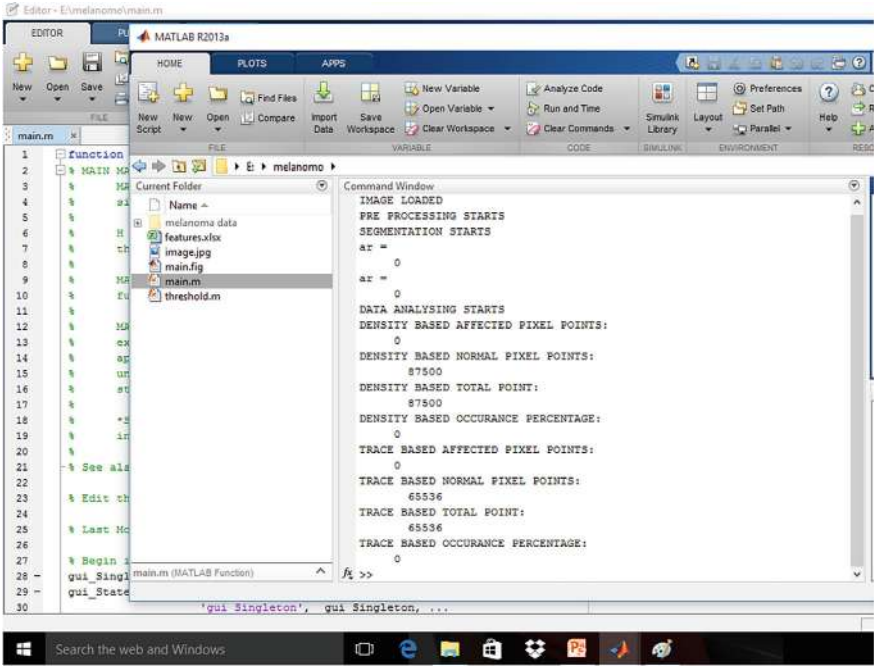


Fig. 5 Command window for affected image

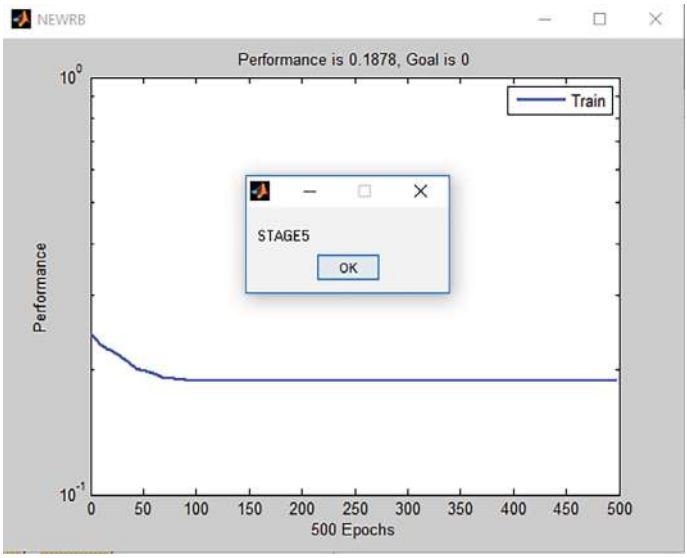


Fig. 6 Graph for affected image

5 Conclusion

The rising prevalence of skin cancer, particularly melanoma, necessitates the development of advanced and reliable diagnostic systems to facilitate early detection and timely intervention. Traditional diagnostic approaches rely heavily on visual inspection by dermatologists, which can be subjective and prone to human error. The integration of Neural Network-Based Smart Detection using Radial Basis Function Networks (RBFN) provides an efficient, automated, and accurate solution for melanoma detection. The proposed system effectively combines image preprocessing, feature extraction, and classification to distinguish between benign and malignant skin lesions. The preprocessing stage, incorporating median filtering, ensures noise reduction and enhances image clarity, while the Radial Basis Function Network (RBFN) algorithm extracts significant texture, color, and shape-based features to improve classification accuracy. RBFNs, known for their fast-learning capabilities and ability to handle nonlinear data, excel in medical image classification tasks, making them highly suitable for real-time skin cancer detection.

The system's high accuracy, as demonstrated by classification results exceeding 99% in some cases, validates its robustness in detecting melanoma. Unlike conventional machine learning techniques, RBFNs leverage radial basis functions as activation functions, offering a smooth and efficient decision boundary. The flexibility of the network in adjusting the number of neurons ensures an optimal trade-off between computational efficiency and classification performance. Additionally, the low Mean Squared Error (MSE) observed across different configurations of the model signifies minimal misclassification, further reinforcing its reliability. Future enhancements could focus on refining artifact removal techniques, improving segmentation accuracy, and incorporating deep learning models to complement RBFNs. The integration of multi-modal imaging, real-time analysis, and cloud-based diagnostics could further expand its practical applicability in clinical settings. By providing objective, consistent, and data-driven diagnoses, the proposed system holds the potential to revolutionize early melanoma detection, ultimately reducing mortality rates and improving patient survival outcomes.

The future scope of Neural Network-Based Smart Detection of Skin Cancer using Radial Basis Function Networks (RBFN) lies in its integration with deep learning models, advanced image segmentation techniques, and real-time deployment for clinical applications. Hybrid approaches combining RBFN with CNNs or SVMs can enhance classification accuracy, while edge computing and cloud-based AI models enable real-time melanoma detection. Further advancements in Explainable AI (XAI) and multi-modal data fusion will improve interpretability and precision. Large-scale clinical validation and regulatory compliance will facilitate widespread adoption, making AI-driven early skin cancer detection more accessible, reliable, and effective in global healthcare systems.

References

1. Abuzaghleh, O., Barkana, B.D., Faezipour, M.: Noninvasive real-time automated skin lesion analysis system for melanoma early detection and prevention. IEEE J. Transl. Eng. Health Med. **3**, 1–12 (2015)
2. Santy, A., Joseph, R.: Segmentation methods for computer aided melanoma detection. In: 2015 Global Conference on Communication Technologies (GCCT), pp. 490–493. IEEE (2015)
3. Joseph, S., Panicker, J.R.: Skin lesion analysis system for melanoma detection with an effective hair segmentation method. In: 2016 International Conference on Information Science (ICIS), pp. 91–96. IEEE (2016)
4. Burada, S., Manjunath Swamy, B.E., Sunil Kumar, M.: Computer-aided diagnosis mechanism for melanoma skin cancer detection using radial basis function network. In: Proceedings of the International Conference on Cognitive and Intelligent Computing: ICCIC 2021, vol. 1, pp. 619–628. Springer Nature Singapore, Singapore (2022)
5. Balaji, M.S.P., Saravanan, S., Chandrasekar, M., Rajkumar, G., Kamalraj, S.: Analysis of basic neural network types for automated skin cancer classification using Firefly optimization method. J. Ambient Intell. Human. Comput. **12**, 7181–7194 (2021)
6. Lakshmi, V.V., Leena Jasmine, J.S.: A hybrid artificial intelligence model for skin cancer diagnosis. Comput. Syst. Sci. Eng. **37**(2), 233–245 (2021)
7. Nyemeesha, V., Mohammed Ismail, B.: A systematic study and approach on detection of classification of skin cancer using back propagated artificial neural networks. Turk. J. Comput. Math. Educ. **12**(11), 1737–1748 (2021)
8. Xu, Z., Sheykhahmad, F.R., Ghadimi, N., Razmjooy, N.: Computer-aided diagnosis of skin cancer based on soft computing techniques. Open Med. **15**(1), 860–871 (2020)
9. Kumar, P., Chauhan, R., Shankar, A., Stephan, T.: Role of artificial intelligence for skin cancer detection. In: Evolving Role of AI and IoMT in the Healthcare Market, pp. 141–174 (2021)
10. Jeny, A.A., Md. Sakib, A.N., Junayed, M.S., Lima, K.A., Ahmed, I., Baharul Islam, Md.: SkNet: a convolutional neural networks based classification approach for skin cancer classes. In: 2020 23rd International Conference on Computer and Information Technology (ICCIT), pp. 1–6. IEEE (2020)
11. Dildar, M., Akram, S., Irfan, M., Khan, H.U., Ramzan, M., Mahmood, A.R., Alsaiari, S.A., Saeed, A.H.M., Alraddadi, M.O., Mahnashi, M.H.: Skin cancer detection: a review using deep learning techniques. Int. J. Environ. Res. Publ. Health **18**(10), 5479 (2021)
12. Roobini, S., Kavitha, M.S., Karthik, S.: A systematic review on machine learning and neural network based models for disease prediction. J. Integr. Sci. Technol. **12**(4), 787–787 (2024)
13. Karpakam, S., Malini, P., Kumar, S., Vijayakumari, G., Thirukkumaran, R., Hanne, L.S.: Deep learning and fuzzy decision support system for visually impaired persons. IEEE Xplore **7** (2022)
14. Chowdary, M.K., Ganesh, C., Michael Preetam Raj, P., Girish Kumar, M., Sridhar, M., Sandhya, S.: An expert system for insulin dosage prediction using machine learning & deep learning algorithms. In: 2023 8th International Conference on Communication and Electronics Systems (ICCES), pp. 1291–1297. IEEE (2023)

Personalized Human Activity Recognition with Transfer Learning

Aniketh Mishra, Namrata Dhanda, and Kapil Kumar Gupta

Abstract Human Activity Recognition (HAR) has been a very dominant domain in pervasive computing and was used to support healthcare monitoring, fitness tracking, and smart environments. Nevertheless, models trained on specific people often malfunction when used on new individuals because actions are not precisely performed differently. To tackle this problem, we present a transfer learning framework geared towards improving the generalization ability of the HAR models for a more far-reaching base of users. To address this problem, we use a publicly available multi-user HAR dataset and design the deep learning architecture of convolutive neural networks (CNNs), and long short-term memory (LSTM) networks to learn spatial temporal patterns from sensor data. Firstly, the model is trained in a source group of users, then it is fine-tuned on some subset of target users with limited (or not for some) labeled data. Experimental results show that our method yields a substantial performance gain when deployed on new users, to up to 12% better than baseline models. The results presented herein illustrate the feasibility of transfer learning in constructing scalable and user independent HAR systems enabling their deployment in the real world.

Keywords Human activity recognition (HAR) transfer learning · Multi-user systems · Deep learning · Sensor data · CNN-LSTM · Domain adaptation · Wearable computing

A. Mishra · N. Dhanda · K. K. Gupta (✉)
Amity University, Lucknow, India
e-mail: kapilkumargupta2007@gmail.com

A. Mishra
e-mail: anikethmishra3415@gmail.com

N. Dhanda
e-mail: ndhanda510@gmail.com

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_3

1 Introduction

Human activity Recognition (HAR) serves a very important role for intelligent application in healthcare monitoring, assisted living, fitness tracking, workplace safety, and smart environments. More specifically, the goal of HAR systems is to automatically detect what activities people are performing, such as walking, sitting, running, etc. from sensor data recorded on mobile devices or wearable technologies. However, there is an increase in the prevalence of smartphones and smartwatches with inertial sensors (accelerometers, gyroscopes, etc.) over the inertial network, which increased the feasibility of scalable, real-time HAR applications.

The efficacy of HAR models has tremendously improved thanks to the introduction of deep learning techniques. Machine learning methods implemented up until now relied mainly on manual feature extraction, challenged by the difficulty adapting to complex or noisy activity data. Deep learning approaches like Convolutional Neural Networks (CNNs) and Long Short Term Memory (LSTM) networks have a highly powerful ability to learn hierarchical features of sensor data directly, and thus learn both the spatial and temporal aspects of the movement patterns of the human.

1.1 Challenges in Multi-user HAR

Although the models of Human Activity Recognition (HAR) made progress, they are concurrent difficulties in their generalization power regardless of how everyone. current HAR systems are developed and assume that sensor placement, activity performance and device orientation are stable in a given cohort. On the other hand, real applications exhibit substantial variability which is inconsistent among users as evidenced by differences in body shape, movement patterns, and sensor placements amongst other things [1].

Models on users who were not involved in the training dataset tend to achieve good recognition accuracy but have a considerable drop when applied to users that vary from those included in the training dataset. However, the approach of retraining the model for new users is unfeasible because of the time, computational resources and data annotation it would imply. It is therefore a critical challenge to meet to enable scalable HAR, because such system must achieve user independent or cross user activity recognition [2, 3].

1.2 Role of Transfer Learning in HAR

Human Activity (HAR) is prone to the cross-user generalization challenge and transfer learning is turning out to be a very helpful approach to overcome it. In other

words, transfer learning consists of leveraging knowledge that has been learned on one domain (source domain), to improve performance on another domain (target domain) when labeled data is scarce. In the context of HAR, it simply refers to training models on sensor data from groups of users, and then fine tuning the models for new users through fine tuning or domain adaptation [4].

Transfer learning replaces these requirements with pre-trained feature representations, limiting retraining large amount of user-supplied data; reduces the amount of time necessary to train the model; and enhances the model's ability to generalize to previously unseen individuals. However, the characteristic is HAR setting where it is often impossible to collect enough labeled data from any user.

1.3 Objective and Contribution

This study proposes a deep learning framework for improving multi-user human activity recognition (HAR) by combing Convolutions Neural Networks (CNN) and Long Short-Term Memory (LSTM) layers. The model is first trained on a group of source users and then fine-tuned on a target group with limited labeled data, simulating real-world application scenarios. The effectiveness of the approach is validated through experiments on a public multi-user HAR dataset.

The key contributions of this work are:

- A hybrid CNN-LSTM architecture is developed for sensor-based Har, capable of capturing both spatial and temporal dependences from multiverse time-series data.
- Transfer learning is employed to allow the model to adapt to new users using only a small amount of labelled data, there enhancing its applicability in practical situations.
- Experiments conducted on a standard multi-user HAR dataset demonstrate that the proposed method outperforms conventional models that do not utilize transfer learning.
- The impact of user variability and the effectiveness of transfer learning on model generalization are thoroughly analyzed, offering insights for building more adaptable HAR systems in the future.

2 Literature Review

2.1 Traditional Approaches to Human Activity Recognition

Human Activity Recognition (HAR) has mainly been approached in the early days with classical machine learning algorithms such as Decision Tree, k-Nearest Neighbors (k-NN), Support Vector Machines (SVM), and Random Forests. They were

typically handcrafted feature models based on those extracted from sensor data—Statistical measures (mean, SD, entropy), frequency domain coefficient (FFT, DCT), domain specific motion features. These were then of good quality and relevant to the problems with hand.

Moreover, common handcrafted features do not generalize well to diverse users and sensor setups, making them fragile in meaningful real application settings. There was also a need for domain expertise and feature engineering that is labor intense. These limitations notwithstanding, however, traditional machine learning served as a good foundation for the study of activity recognition and its relevance to resource constrained or interpretable systems [5].

2.2 Deep Learning in HAR

As feature engineering is fundament constraint in traditional methodologies, hence it is in the interest of developing deep learning models capable of endowing themselves of hierarchically informed representations learned from the raw sensor data. A variety of techniques have employed convolving neural network (CNNs) to extract spatial features from the times series segment of accelerometer and gyroscope information. Often, CNNs are combined with the Curved Neural Network in the form of DeepConvLSTM and the CNN-LSTM to enhance performance, taking advantage of the localized signal characteristics but preserving the temporal relationships.

In addition, Recurrent Neural Networks (RNN) such as the Long Short-Term Memory (LSTM) networks have been quite popular as they are good at modelling sequential dependencies in sensor data streams. LSTM networks are also very good at spotting for continued or overlapping activities. Hybrid models of CNNs for spatial extraction features with LSTMs for temporal analysis have been introduced through a number of studies in numerous instances which have resulted in increased accuracy of classification on benchmark datasets such as UCI HAR, WISDM and PAMAP2.

Yet, although they have achieved this, deep learning models often exhibit user specific performance, performing well for users in the training phase, while they struggle in generalizing new users due to differences in sensor placements, occasional variations in the way activity is executed, and varied levels of noise.

2.3 Multi-User HAR and Cross-User Generalization

Aware of the importance of model generalization, researchers also try to build models that perform well with the users who are never seen before. Because individuals are more variable, it is harder to achieve cross user generalization than to use user dependent models. A generalization capability is given by one proposed method of consolidating data from the quantity of users spanning across wide range during the

training phase. However, acquiring such datasets, which must be extensive and well annotated with many participants can be costly and labor intensive.

Specifically, several studies have attempted to alleviate this problem by exploring domain adaptation procedures for transforming feature representations from one user domain into another. This is done through MMD based methods or adversarial training including Domain Adversarial Neural Networks or instance reweighting. Their attempts to develop a domain invariant representation that is still robust to user specific variations. However, these methods are promising yet often involve expensive training procedures and can still rely on labeled data from the target domain.

2.4 Transfer Learning in HAR

Dominant approaches in the area of domain adaptation have been transfer learning that is more adaptable and scalable. Transfer learning is typically discussed in the context of Human Activity Recognition (HAR) where a model is pre-trained on a source domain composed of a set of users (cohort) or publicly accessible dataset and then fine-tuned on a small number of labeled subsets of the target domain (the ones not seen before by the model), i.e. the new users. It allows us to minimize the need for large, annotated datasets and leverage the learned representations in other related domains.

However, many of the studies on transfer learning in HAR have been successful. For instance, Hämmerli et al. showed how adjustments deep models can be performed across different sensor modalities using transfer learning. They also studied representation transfer through unsupervised domain adaptation and fine-tuning methods as presented in Guan and Plötz, 2021. In these investigations, substantial improvements over cross-user performance were made with orders of magnitude less target domain training data.

Recent reports also include layer freezing techniques, where the lower layers or feat32 re extraction layers stay fixed, while the upper layers, designed specifically for a task, are trained for new users. With regards to transferable knowledge and accommodating user specific characteristics, this strategy facilitates.

However, work on user-scalable, low data HAR systems still remains, uncompleted. Transfer learning research in the HAR domain is still fueled by ongoing challenges in terms of data heterogeneity, personalization and sensor variability. Although deep learning has made considerable progress in human activity recognition (HAR), its dependence on training data specific to individual users restricts its effectiveness in real-world scenarios. Traditional methods struggle to accurately model intricate sensor patterns, and many deep learning approaches still exhibit a lack of robustness when applied to a diverse user base. Transfer learning presents a promising approach by facilitating adaptation across different users with minimal data; however, its application in HAR remains an ongoing research focus [2].

This research seeks to address this limitation by creating a HAR framework based on transfer learning that integrates the advantages of convolutional neural networks (CNNs) and long short-term memory networks (LSTMs). Our approach distinguishes itself from prior studies by concentrating specifically on multi-user adaptation, utilizing a publicly available dataset, and showcasing tangible improvements in generalization through well-structured transfer learning methodologies.

3 Methodology

This thesis describes the resultant extensive methodology used towards the creation and assessment of a multiuser Human Activity Recognition (HAR) transfer learning framework. We present our approach which includes a carefully picked and tuned dataset, as well as a hybrid deep learning model with proper transfer learning, be used to guarantee strong generalization for diversified users but also little data adaptation.

The suggested method recognises human activity from sensor data using a hybrid CNN-LSTM model. By fine-tuning the model on new users with less labelled data after training on existing users, it uses transfer learning to increase accuracy and generalisation.

3.1 Dataset Description

For this research, the Wireless Sensor Data Mining (WISDM) v 1.1 data set was used, that can be found on Kaggle. This dataset is freely available to aid as a public benchmark for human activity recognition (HAR) using sensor data and is used in many academic studies due to extensive diverseness, high temporal resolution, and high accuracy to real world activities. From 36 participants who held Android smartphones in their front pocket of their pants, data was gathered. Six different physical activities were performed by these people—walking, jogging, sitting, standing, climbing stairs and descending stairs. The data sample is tri-axial accelerometer measurements at 20 Hz frequency, which provides precision in the temporal sampling of motion data.

The dataset encompasses roughly one million time-stamped sensor readings. The user ID, timestamp, x/y/z acceleration values and the activity label matching that value are what each entry contains. The amount of each participant's activity data is approximately 30 min of uninterrupted data, providing an optimal balance in the scope of physical motion patterns across the sample population. Users in this dataset move with significantly different style, different sensors were used and different execution speeds, making this dataset very fitting for user independent HAR system development and evaluation.

3.2 Data Preprocessing

A comprehensive preprocessing pipeline was then established for preparing the raw accelerometer data for input to the deep learning model. We divided the continuous time series signals into fixed length overlapping segments of size five seconds and using 50% overlap. The continuous data are segmented into discrete samples and providing enough temporal information so that elements within the samples can differentiate between various activities.

Normalizing the accelerometer data withing each segment by z score standardization then gets rid of amplitude biases among users because of different proportions of movement intensity and body composition. Different channels were kept separate on the three axes of acceleration to conserve spatial information. The data was structured to maintain user identities so that it could be split in a controlled source target manner and activity labels were converted into categorical identifiers in order to support supervised learning [6].

Users were also categorized into two different groups: source and target users. Source users are ones whose data was used to train the model initially, and target users are instance that have their data used for fine tuning and some for evaluations purposes. This arrangement captures the real world setting whereby the model ought to update itself to new users with very little labeled data.

3.3 Model Architecture

In the proposed approach, a hybrid deep learning architecture, wherein the Convolutional Neural Networks (CNNs) with Long Short Term Memory (LSTM) units were exploited. This design aims to leverage both spatial relationships among sensor axes and temporal variations over time. Three channels of each window are formatted as a matrix containing the x, y, and z accelerations of sensor data during the time series [7] (Table 1).

Table 1 Summary of the WISDM v1.1 dataset

Attribute	Description
Number of users	36
Sampling frequency	20 Hz
Sensor type	Tri-axial accelerometer
Activities recorded	walking, jogging, sitting upstairs, downstairs
Total data points	~ 1 million
Averages durations/user	~ 30 min
Data format	Timestamp, User ID, x/y/z, acceleration

The convolutional layers here act as spatial feature extractors. The first convolutional layer has 64 filters (kernel size of five) to capture localized variations in the signal and does a max pooling to reduce dimensionality and enhance translational invariance. Still, with a subsequent convolutional layer, this consists of 128 filters with a smaller kernel size of three for refining the extracted features. To combat the overfitting issue, a dropout layer is included after the convolutional layers to randomly deactivate some neurons when training.

The output from CNN is reshaped and input into LSTM layer with 100 hidden units. This recurrent layer is good at capturing sequential dependencies and temporal patterns that are important for identifying types of activities such as walking and jogging that are dynamic [8].

3.4 Transfer Learning Strategy

Consequently, the structured transfer learning approach was carried out in two stages, namely pretraining and fine-tuning, to achieve user independence recognition. In the pretraining stage, the hybrid model was trained with labeled data of source user group. In this phase, the model could learn generalizable spatial temporal activity patterns to be applied in different individuals. The training was done with Adam optimizer with learning rate to 0.001 using categorical cross entropy loss function.

After that, the pre-trained model was fine-tuned by using limited labeled samples belonging to the target users at the fine-tuning stage. Two adaptation methods were explored. The first method was the fine tune of all layers of the model with the new data, which completely fitted it to learn the user specific characteristics. The second method is used to freeze the convolutional layers while updating only LSTM and output (Table 2).

The layers where we presumed that spatial features are more universal whereas temporal dynamic could be user specific. Freezing layers help mitigate overfitting

Table 2 Architecture of the proposed CNN-LSTM model

Layer	Configuration	nFunction
Input	(Window size × 3)	Time-series input from x, y, z acceleration
1D convolution	64 filters, kernel size = 5	Local feature extraction across time
Max pooling	Pool size = 2	Temporal down sampling
1D convolution	128 filters, kernel size = 3	Higher-level feature learning
Dropout	Rate = 0.3	Regularization to prevent overfitting
LSTM	100 units	Sequential modeling of temporal dependencies
Fully connected (FC)	Dense layer	Feature integration
Output	SoftMax', 6 units	Multiclass activity prediction

and shortens the training time, which makes this a good strategy for applications such as the real time or resource restricted problems [9].

The model was trained in the target user data and its performance in classifying the reserved segment of the target user data was used to analyze the effectiveness of the transfer learning process. Finally, comparison to baseline models that have been trained from scratch on the same limited dataset, and comparison to models that are not fine-tuned for target users, are made.

3.5 Experimental Setup

Three different experimental conditions were established to measure the effectiveness of the proposed framework. The first condition was training and testing a baseline model with data from the same users—this upper limit comes from a user dependent context. The second condition involved testing a model that was trained on source users on target users without modification, i.e., zero-shot, in order to simulate such a deployment. In the third condition, a transfer learning approach was adopted, in which we fine-tuned the pre trained model with a bounded dataset from the target users [10].

Metrics such as accuracy, precision, precision recall, F1-score etc. were used to evaluate performance. For class imbalance, only these metrics were calculated for each class, and then averaged with macro and weighted methods. All experiments conducted in multiple user splits for robustness, and statistical reliability [3].

4 Results and Discussion

An empirical assessment of the suggested transfer learning framework for multiuser Human Activity Recognition (HAR) is provided in this section. To evaluate the factors that determine inter user variability, we run a series of experiments to compare user dependent, user independent and transfer learning methods, on their merits and demerits. Our analysis encompasses performance metrics, interpretations of the confusion matrix, and reflections on the trade-offs between generalization and personalization.

4.1 Evaluation Metrices

For thorough performance evaluation of the model we used multi class classification metrics such as, accuracy, precision, recall and F1 score. They provide an overall and the quality of each class [11]. As HAR (human activity recognition) datasets often feature natural class imbalance of dynamic activities (such as walking) that occur

for longer periods of time than short occurrences of stair climbing or transitional postures, we calculated both macro and weighted average. Moreover, it also averaged the metrics across different user splits to improve statistical reliability.

4.2 Baseline Comparison

To illustrate the advantages of transfer learning, we initially developed two baseline models. The user-dependent model, which was both trained and evaluated using data from the same individual, demonstrated superior performance; however, it was limited in its ability to generalize. Conversely, the user-independent model, which was trained on a group of source users and tested on previously unseen target users without any adjustments, experienced a notable decline in performance. This decline can be attributed to the variability in motion patterns and device positioning among different users. This finding highlights the challenges associated with achieving generalization across users in human activity recognition (HAR) [12].

4.3 Transfer Learning Performance

For the proposed framework, the CNN-LSTM model first is fine-tuned, using a small dataset from the target users, and then learned over new users' sequences. Two adaptation strategies were evaluated: complete model fine tuning and partial fine tuning; that is, freezing the CNN layers and adapting the rest of the model. Based on both methods, the performance has improved greatly than user-independent baseline and complete fine-tuning method always outer-forms the others across all evaluation metrics.

The highest average classification accuracy (74.21%) (and corresponding F1-score of 0.732) was improved to 89.34% (0.891) when full fine tuning was used. Moreover, these results prove that the model is capable of adapting to new users with very little labeled data [13]. These results suggest that spatial temporal features learned from source users can be utilized and refined to be useful to the target users, that have their own motion pattern.

Finally, Fig. 1 further illustrates these advancements by showing comparative learning configurations within user dependent, user independent, and transfer learning on their classification accuracy and macro averaged F1-score. This visual representation shows the extreme difference in performance that this gulf of generalization and personalization' has when we are on user-independent models and fine-tuned models, which shows how transfer learning closes this gap.

Figure 1 comparison of accuracy and F1-score of users dependent, user independent, transfer learning models. What the findings show is how transfer learning can close the performance gap without having to gather large amounts of user specific data.

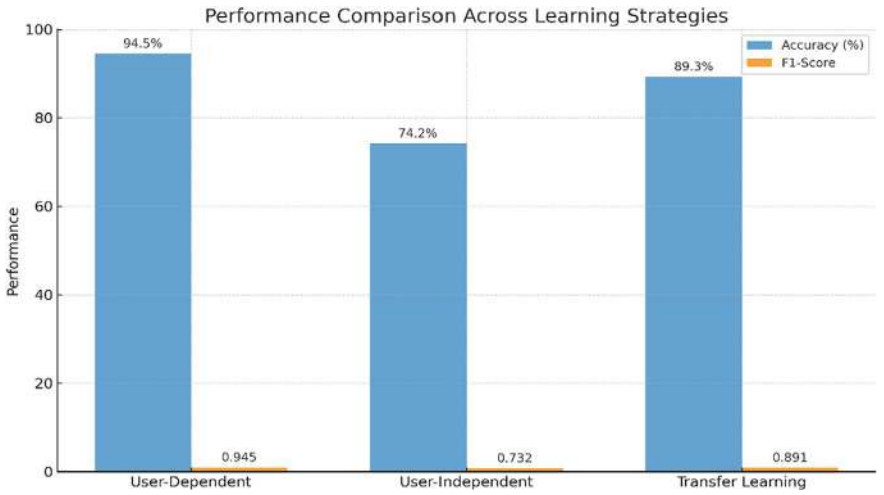


Fig. 1 Performance comparison across learning strategies: user-dependent, user-independent, and transfer learning, evaluated by accuracy and F1-score

4.4 Activity-Wise Analysis

An analysis of activity levels is done in detail by identifying dynamic actions namely Walk, Jog and Stair Descent, more accurately than static positions such as Sitting or Standing. This observation is consistent with previous findings of Human Activity Recognition (HAR) and can be justified by the particular acceleration and orientation patterns associated with dynamic movements. In addition, the use of transfer learning helped in not confusing similar categories. Particularly in the user independent case, significant enhancements were noted in distinguishing Sitting from Standing, which is often a very common mistake [14].

4.5 Confusion Matrix Insights

In analyzing class-wise performance improvements, confusion matrices were examined before and after the fine-tuning process. The pre-adaptation matrix showed common mistakes between Standing and Sitting as well as between Walking and Upstairs which implies that more specific features are needed beyond basic user-shared archetypes models. After the application of fine-tuning, these misclassifications were greatly improved (see Fig. 3), confirming that transfer learning enabled the model, to a greater extent, accurately capture individual biomechanical patterns and transitional movements.

Figures 2 and 3 matrices of confusion for before and after fine-tuning. The fine-tuned models demonstrate increased dominance of diagonals alongside diminished confusion between classes, particularly for posture-based activities.

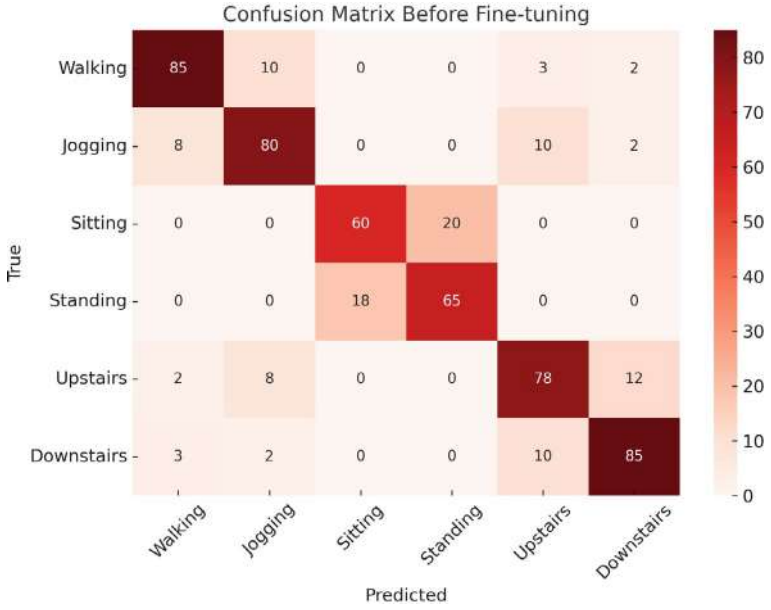


Fig. 2 Confusion matrix before fine-tuning

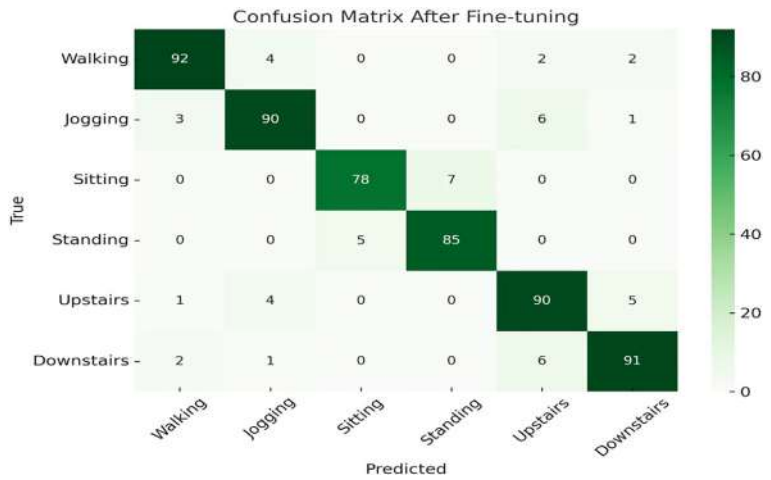


Fig. 3 Confusion matrix after fine-tuning

4.6 Discussion on Generalization and Scalability

The findings emphasize two aspects of the proposed transfer learning approach: customization optimization for individual users while generalization across users is preserved. Such scalability is important for real-world human activity recognition (HAR) systems because, due to privacy, financial, and time limitations, collecting copious amounts of labeled data for each person is impractical [15]. Our framework lessens these burdens by only needing a small amount of fine-tuning data from the user.

Moreover, the architecture's building blocks, which include CNN's spatial abstraction and LSTM's capture of temporal dynamics, worked well for HAR. Retaining lower layers during partial unfrozen adaptation to refined upper layers helped in the swift fine-tuning. This effective learning strategy provides a streamlined approach that is conducive to real-time application in smart homes, healthcare monitoring, fitness tracking, and sophisticated AI systems, resulting in improved computational efficiency.

5 Conclusion and Future Scope

The research developed a powerful and flexible transfer learning framework that attempts to solve problems with multi-user Human Activity Recognition (HAR) systems. Using a hybrid CNN-LSTM model, the architecture spatially and temporally retrieved data using wearable sensors. Also, the transfer learning methods facilitated the new user's incorporation with little extra training data.

The experiments demonstrated that most user-independent models performed poorly in cross-user scenarios. This is frequently caused by differences in dynamics of the body, the way the sensors are placed, and behavioral patterns. These results exemplify the effectiveness of transfer learning as a means to improve performance for the given model, with the proposed model benefitting the most from transfer learning, achieving accuracy from 74.21 to 89.34% and macro F1 score from 0.732 to 0.891. The model proposed proved that transfer learning is helpful to narrow the gap between user-independent models and user-dependent models with less data and effort needed to collect the data.

Additional to strong quantitative scores, qualitative insights from the confusion matrix also underlined the model's competencies in resolving ambiguities across similar classes, for instance, sitting vs Standing and Walking vs Stair Climbing. Also concerning system architecture, its modular design permits both full and partial fine-tuning, which is beneficial for scaling to more resource-constrained platforms like smartphones or embedded IoT devices.

Looking ahead, there are other impactful ways this work can be further developed and advanced. Future work may look into unsupervised domain adaptation, so no

labelled target user data is required for generalization, enabling zero-shot generalization. Federated learning frameworks could be added to facilitate personalization on the device while protecting user identity. Attention mechanisms, self-supervised training strategies, or transformer-based methods could boost the model's ability to capture long-range activity-contextual dependencies and activity sequence nuances. Lastly, extending the model to multi-sensor fusion support, for example through combining accelerometers, gyroscopes, and environmental sensors, could improve its performance in dealing with diverse real-world scenarios.

As noted, the proposed framework based on transfer learning offers an efficient, accurate, and cost-effective solution to multi-user activity recognition systems, enabling further development toward intelligent and user-centered computing technologies.

The suggested method effectively illustrates how transfer learning may bridge the gap between user-dependent and user-independent models with little more data, greatly increasing the accuracy and flexibility of HAR systems.

References

1. Bursa, S.O., Incel, O.D., Alptekin, G.I.: Personalized and motion-based human activity recognition with transfer learning and compressed deep learning models. Comput. Electr. Eng. **109**, 108777 (2023)
2. Ferrari, A., et al.: Deep learning and model personalization in sensor-based human activity recognition. J. Reliab. Intell. Environ. **9**(1), 27–39 (2023)
3. Amrani, H., Micucci, D., Napoletano, P.: Personalized models in human activity recognition using deep learning. In: 2020 25th International Conference on Pattern Recognition (ICPR). IEEE (2021)
4. Dhekane, S.G., Ploetz, T.: Transfer learning in human activity recognition: a survey. arXiv preprint arXiv:2401.10185 (2024)
5. An, S., et al.: Transfer learning for human activity recognition using representational analysis of neural networks. ACM Trans. Comput. Healthc. **4**(1), 1–21 (2023)
6. Ahmad, W., Kazmi, M., Ali, H.: Human activity recognition using multi-head CNN followed by LSTM. In: Proceedings of the International Conference on Artificial Intelligence, pp. 1–7 (2020).
7. Dhekane, S., Deisenroth, M.P., Roggen, D., Ploetz, T.: Transfer learning in sensor-based human activity recognition: a survey. ACM Comput. Surv. **56**(3), 1–35 (2024)
8. Oluwalade, B., Neela, S., Wawira, J., Adejumo, T., Purkayastha, S.: Human activity recognition using deep learning models on smartphones and smartwatches sensor data. arXiv preprint arXiv:2103.03836 (2021)
9. Raj, R., Kos, A.: An improved human activity recognition technique based on convolutional neural network. Sci. Rep. **13**(1), 22581 (2023)
10. Yuan, H., Chan, S., Creagh, A.P., Tong, C.: Self-supervised learning for human activity recognition using 700,000 person-days of wearable data. NPJ Digit. Med. **7**(1), 91 (2024)
11. Rokni, S.A., Nourollahi, M., Ghasemzadeh, H.: Personalized human activity recognition using convolutional neural networks. Proc. AAAI Conf. Artifi. Intell. **32**(1) (2018)
12. Hernandez, N., et al.: Literature review on transfer learning for human activity recognition using mobile and wearable devices with environmental technology. SN Comput. Sci. **1**(2), 66 (2020)

13. Bettini, C., Civitarese, G., Presotto, R.: Personalized semi-supervised federated learning for human activity recognition. arXiv preprint arXiv:2104.08094 (2021)
14. Lin, C.-Y., Marculescu, R.: Model personalization for human activity recognition. In: 2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops). IEEE (2020)
15. Alawneh, L., et al.: Personalized human activity recognition using deep learning and edge-cloud architecture. J. Ambient Intell. Human. Comput. **14**(9), 12021–12033 (2023)

Efficient Handwritten Text Recognition Using Residual Networks and BiLSTM

G. Rakshitha, I. M. Rozana, Rohit B. Patil, and Pooja Shrivastav

Abstract Recognising handwritten text is essential to automating a number of real-world tasks, including data entry, archive procedures, identity verification, and document digitisation. The variety of writing styles, inconsistent spacing, and distortions in handwritten inputs make it difficult to accurately translate handwritten content into machine-readable text. This research presents a deep learning-based system that integrates Recurrent Neural Networks (RNN), Bidirectional Long Short-Term Memory (BiLSTM), and Connectionist Temporal Classification (CTC) to tackle these issues. For the purpose of deciphering intricate sequences in cursive or unstructured handwriting, the RNN-BiLSTM framework allows the model to efficiently capture contextual relationships in both forward and backward temporal directions. Character segmentation during training is no longer necessary because of the incorporation of CTC decoding, which enables the model to learn from unsegmented sequences. To enhance the model's generalisation across various writing styles, the system is trained using the IAM Handwriting Dataset, which comprises a diverse collection of natural handwriting examples. The suggested approach shows enhanced recognition accuracy and is particularly well-suited for real-world applications in fields like legal documents, secure identity systems, and extensive digitisation projects that demand accurate and effective text recognition.

Keywords Handwritten text recognition (HTR) · Residual neural networks (ResNet) · Connectionist temporal classification (CTC) · Data augmentation · Deep learning · Optical character recognition (OCR) · Supervised learning

G. Rakshitha · I. M. Rozana · R. B. Patil · P. Shrivastav (✉)
Department of MCA, CMR Institute of Technology, Bengaluru, India
e-mail: pooja.s@cmrit.ac.in

G. Rakshitha
e-mail: rag23mca@cmrit.ac.in

I. M. Rozana
e-mail: roim23mca@cmrit.ac.in

R. B. Patil
e-mail: roba23mca@cmrit.ac.in

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_4

1 Introduction

The evolution of Handwritten Text Recognition (HTR) has its roots in the early development of Optical Character Recognition (OCR) systems during the mid-twentieth century to begin with, OCR systems only attempted to record a very specific range within the borders of a language or a font type. The ability to identify particular character sets for different languages was later added. But for these systems, handwriting and all its variations were difficult due to the wide range of differences in letter spacing, writing styles, lines, and other complex concerns [1]. The banking, legal, educational, and health industries had to digitize paper records immediately, so as reliance on technology increased, focus moved to developing methods for decoding handwritten writings. Conventional techniques such as handmade features, template matching, and rule-based algorithms are useless because they cannot manage the large number of different handwriting styles [2]. A major breakthrough in this field was brought by machine learning [3] and deep learning techniques [4]. Scientists later put Convolutional Neural Networks (CNNs) and Residual Neural Networks (RNNs) to use taking advantage of residual connections. This boosted how well deep learning models worked and made a big difference in how they could pull out important features [5]. Residual connections have made it easy to train deep networks. They allow gradients to flow through the model without trouble and work well in any situation. Patterns can be spotted in complex text formats, like handwritten documents [6]. Methods based on Connectionist Temporal Classification (CTC) have revolutionized the decoding process. CTC's ability to predict sequences of different lengths without splitting characters is a big plus for tasks that involve recognizing handwritten text [7]. In this paper, we present a comprehensive online handwritten recognition system that integrates Reset for robust feature extraction and CTC for efficient sequence decoding. The system utilizes the IAM Handwriting Dataset for training which is a well-known standard for handwriting recognition [8].

The workflow consists of data preprocessing such as resizing, normalization, and augmentation, model training employing other optimization techniques like early stopping and adjusting the learning rate, and the inference phase where the CTC decoding algorithm is applied to transcribe the handwritten output into a readable format text.

2 Related Work

Recognizing handwritten text has been at the forefront in the digitization of documents especially in creating machine-readable text from handwritten notes, historical materials, and forms. A wide variety of approaches from standard pattern recognition to sophisticated deep learning techniques have been researched towards improving the effectiveness of handwriting text recognition systems.

A. Traditional Handwritten Text Recognition Techniques

The early stages of OCR relied heavily on character recognition using template matching and statistical character recognition methods. These techniques generally worked well with printed texts, but there were so many problems in detecting handwritten texts because of the differences in a person's handwriting, poor character spacing, and distortion.

Katoch et al. [2] discussed the various challenges associated with traditional methods in their study. Traditional methods included hand crafted features extraction methods including zoning, projection histograms, and contour analysis to represent handwritten characters.

Feature-Based Approaches

Approaches Based on Features Specialized techniques like zoning, projection histograms and contour analysis were used to capture handwritten characters. Classifiers such as K-Nearest Neighbours (KNN) [9] and Support Vector Machines (SVM) [10] were used for grading. Justice had been done but it was noted that a lot of steps had to be taken before the grades were given and it was very weak to noise and distortion so this system was very ineffective when it came to large dodge data sets [11].

B. Machine Learning Approaches

Prior to some of the first end-to-end contemporary deep learning models, Lecun et al. concentrated on applying gradient-based learning strategies with multi-module machine learning models [12]. The application of a Hidden Markov Model (HMM) for the OCR problem was the next significant advancement in achieving high OCR accuracies. HMMs used for planar biometric signature verification were trained with sets of static data. Individual letters of the alphabet or elements of a sign were hypothetically randomly ordered by their so-called pre-established succession [13]. These scripts produced reasonable results. The methods, however required manually extracted features which could not handle the complexity and variety of handwriting.

Before deep learning there was the use of shallow neural networks for HTR like Multilayer Perceptron's (MLP) [14]. These models obtained a reasonable amount of success however they tend to ignore spatial relationships associated with elaborate handwriting which led to the need for very sophisticated manual feature extraction approaches along with very sophisticated pre-processing [5]. RishiKeshan et al. [7] proposed handwriting recognition models from start to finish using Connectionist Temporal Classification (CTC) without needing pre-cut letter labels. This method boosted recognition accuracy on tricky handwriting datasets letting models predict whole words or sentences without lining up each letter.

C. Deep Learning Techniques in Handwritten Text Recognition

CNNs significantly advanced the automated recognition of handwriting through their astonishing ability to learn to understand the location of features within information

displayed as pixels. Research using datasets such as MNIST and IAM demonstrated significant improvements in recognition accuracy with CNN-based models. Lekshmy et al. [15] used CNN for OCR in exam reader system. As opposed to other complex algorithms, CNNs were able to effortlessly derive hierarchical features from images aiding in the automated recognition of varying handwriting. To handle sequential dependencies in handwritten text, a hybrid method including RNNs and Long Short-Term Memory (LSTM) was proposed by Dutta et al. [16]. These models worked well in spotting cursive and joined-up handwriting where letter edges weren't marked. Hemanth et al. [17] used CNN, LSTM and CTC for handwritten text recognition and obtained an accuracy of 98%. Muthukumar et al. presented a hybrid model approach in [18] that combines Pixel Shifting Optimisation and LSTM networks to enhance handwritten character recognition. This hybrid strategy improves the model's capacity to correctly anticipate text from photos by refining stroke prediction. Mixing CNNs to pull out features and LSTMs to model sequences became a common method in HTR studies. Due to their unidirectional LSTM, these methods struggle with complex handwriting and have limited context capture.

The proposed BiLSTM + CTC improves overall identification accuracy by processing sequences in both directions and doing away with the necessity for pre-segmentation.

3 Methodology

A pipeline for handwritten text recognition is shown in Fig. 1.

It starts with a handwritten image as input, which is preprocessed using grayscale conversion, binarization, noise reduction, and normalization. Following preprocessing, the image is segmented to identify pertinent text areas, and then a CNN architecture is used to extract features. ResNet is used to improve representation of these features further. To process sequential and variable-length text recognition, the

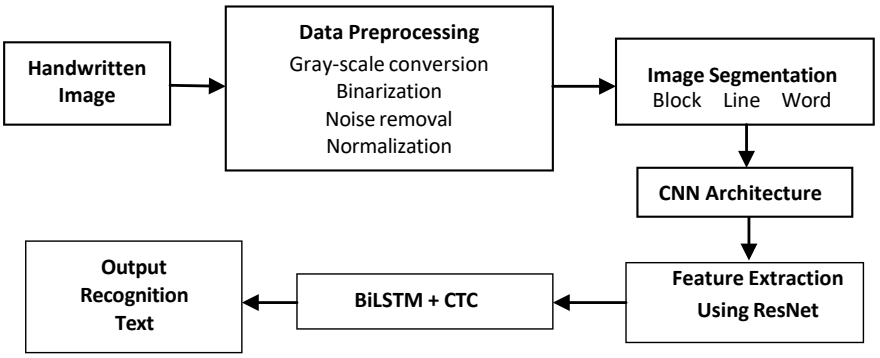


Fig. 1 System architecture

extracted features are run via a BiLSTM network in conjunction with a Connectionist Temporal Classification (CTC) layer, producing the final output text.

The process consists of six main stages as shown in Fig. 2:

A. Dataset Acquisition and Preprocessing—Preparing and augmenting the IAM Handwriting Dataset.
B. Feature Extraction using Residual Neural Networks (ResNet)—Capturing spatial features from images.
C. Sequence Modeling using Connectionist Temporal Classification (CTC)—Decoding text from variable-length sequences.
D. Model Training and Optimization—Training using TensorFlow with various augmentation techniques and callbacks.
E. Inference and Evaluation—Converting new images into text and evaluating accuracy.

A. **Dataset Acquisition and Preprocessing**

The IAM Handwriting Dataset, which contains handwritten English text samples, is used to train the handwriting recognition system. This is actually the pre-processed data for training to substantiate improvement in speed and accuracy of the model. First, the photographs of handwritten texts are converted into Gray images in order to retain important details. This step eliminates excess colour details. The images are then resized, usually to 128 by 32 pixels. This imparts every image in the input the same shape, which helps the model in handling inputs more efficiently. The pixel values are scaled down to range between 0 and 1. This stabilizes the training and prevents very high values from hindering the learning performance of the model. Label encoding is then applied to transform textual data into numerical sequences by assigning numerical labels to assist in supervised learning. These preparatory and

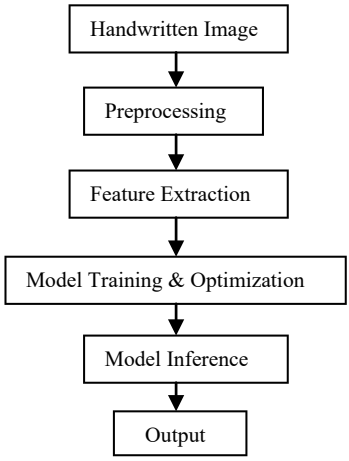


Fig. 2 The work flow of handwriting recognition

enriching processes together ensure that the results are more accurate and dependable, and thus more reliable in actual use.

B. **Feature Extraction Using ResNet**

The Residual Neural Network (ResNet) architecture plays a key role in extracting essential features from handwritten text images. traditional deep networks suffer from vanishing gradients, which hinder the training of deep models. To address this issue, ResNet employs skip connections, allowing the gradient to bypass some layers via backpropagation. This ensures that crucial spatial features are retained across multiple layers, facilitating deeper network structures without sacrificing performance [19].

The network begins with convolutional layers, which identify fundamental features such as lines, edges, and character contours. These extracted features are then processed through residual blocks, which help preserve essential information without degradation. Batch normalization maintains consistent activations within layers, leading to more stable convergence during training.

Additionally, dropout regularization is applied to prevent overfitting by randomly disabling neurons, enabling the model to generalize well across various handwriting styles. The combination of residual learning, convolutional feature extraction, and regularization techniques ensures that the system effectively captures diverse handwriting styles with both stability and accuracy.

C. **Sequence Modeling Using BiLSTM and CTC**

Once important features have been identified they are put into a Bidirectional Long Short-Term Memory (BiLSTM) net which aids in the understanding of the order of the characters within the handwritten text. A standard LSTM network reads the sequences in only one direction; whereas, a BiLSTM reads the text in both directions (forward and backward). The network can better capture contextual dependencies, which can be advantageous when recognizing running script and cursive because the letters do not have as clear of boundaries as standard writing [20].

In order to convert the predicted text from the model to a classification, it uses a powerful framework called Connectionist Temporal Classification (CTC). CTC minimizes the predicted text to the original text at the character level. This makes it easy for the model to correctly classify variable-length text sequences while efficiently removing duplicates and terrible spacing issues [21]. For example, when the model outputs “h e e l l o _”, the CTC decoding conversion will alter this to become “hello”—ensuring that the predicted text produces correct and readable results.

By fusing sequence modeling through BiLSTM and decoding with CTC, the system efficiently and accurately performs the transformation of unstructured handwritten text to well-structured digital text. This approach enables the model to efficiently accommodate varying input lengths and adapt to a variety of handwriting styles.

D. Model Optimization

To increase efficiency and prevent overfitting, the model is improved using a variety of strategies after being trained in TensorFlow. The training program is developed to support robust learning and preserve generalization. The model uses the CTC loss function to solve poor text alignment across sequence-based tasks. Adam (adaptive moment estimate) is employed as the optimizer to get the best learning efficiency [22]. Compared to baseline optimizers like SGD (stochastic gradient descent), Adam evaluates the prior gradients to deliver an enhanced learning rate, which significantly speeds up and smoothes training. By allowing the model to learn sensibly and avoiding overlearning, Adam allows the neural network to converge appropriately without becoming stuck in local minima. During training, the Adam optimizer adjusts itself with a learning rate of 0.0005. For optimal efficiency and computational stability, a batch size of 16 pictures is utilized. Numerous tactics are used to increase the effectiveness of training. Thus, early stopping prevents overfitting and conserves resources by tracking validation loss during training and interrupting the process after performance does not improve for five consecutive epochs. Tensor Board logging provides visual feedback on training progress, learning rate fluctuations, loss curves, and precise trends.

In addition to allowing the model to be tailored for various text formats and handwriting styles, this combination of optimization techniques will increase the model's accuracy, stability, and efficiency.

E. Inference and Evaluation

Following training, fresh handwritten text image data samples are utilized to evaluate the model's performance and accuracy. This well-defined inference phase yields accurate and reliable predictions. To improve the model's predictions, pixel values in the input photos are normalized and reduced to a predetermined size. In the preprocessing stage, the ResNet-based feature extractor is responsible for identifying pertinent spatial patterns in the handwriting. After receiving the characteristics, the BiLSTM-CTC decoder uses learnt patterns to predict the text sequences [16]. The outcomes of the model are assessed using a variety of performance criteria. The character error rate (CER), a quality metric at the letter level, quantifies the model's ability to identify individual letters. Word reconstruction is ensured by the system itself, and word accuracy rate, or WAR, is concerned with accounting for the quantity of words properly predicted. Additionally, the speed of inference of the model is examined to ascertain how quickly it can identify text in real time. The model's ability to recognize realistic and useful handwriting is then confirmed by this method's evaluation modes, although it would struggle to do so in challenging situations.

4 Datasets

High-quality datasets are required for any modelling, training and learning. These figures are made up of handwritten samples gathered in various locations to allow the models to learn and adapt towards different handwriting styles for better accuracy. Popular datasets include the RIMES, CVL, IAM Handwriting Database, and MNIST. The famous MNIST database is almost always used for digit classification applications and contains almost all handwritten digits from 0 to 9. The IAM Handwriting Database contains millions of handwritten words, lines, and even sentences and paragraphs that form the corpus for recognizing handwritten English text. Moreover, because RIMES and CVL databases can identify handwritten texts in more than one language, they are particularly helpful for any multi-lingual studies and also for the research focused on particular scripts and writing systems.

The IAM Handwriting Database [23] is an important part of the datasets necessary for testing and training handwriting recognition models. It provides models with exposure to a vast variety of writing styles through handwritten English sample materials from many writers. 353 handwritten English text images with a PNG extension and matching text files with the TXT label are included in the dataset. Each dataset entry has a fixed form, a unique image ID, a status indication, the sample quality, the bounding box coordinates that show the location of the handwritten text, and the associated ground-truth label. To offer high-quality handwriting recognition, writer identity, and verification test data, images in the IAM dataset are scanned at a resolution of 300 dpi and saved in 256 levels of grayscale.

In the given Fig. 3 the word 'activity' is captured as illustrated; it is shown during freehand writing with some variability in stroke weight and alignment. People's handwriting varies from one person to the other and this variety is a challenge to recognition models.

But deep learning-based recognition systems trained on vast handwriting databases such as IAM are aimed at handling these changes effectively. These systems extract significant data from images and employ sophisticated techniques to transform handwritten text into machine-readable text. With these statistics and deep learning techniques, contemporary handwriting recognition systems have had high

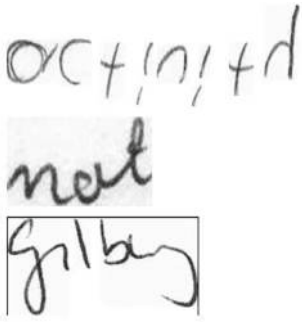


Fig. 3 Example image from IAM dataset [23]

precision and robustness in dealing with different handwriting styles. The dataset consisted of 13,353 handwritten text images, which were pre-processed by resizing, normalizing, and enhancing contrast to improve feature extraction. The images were converted to grayscale and fed into the ResNet-BiLSTM-CTC model.

Each line in the words.txt file includes metadata with the following format:

a01-000u-00-00 ok 154 408 768 27 51 408 768 THE
a01-000u-00-01 ok 156 507 768 27 51 507 768 QUICK
a01-000u-00-02 ok 156 604 768 27 51 604 768 BROWN.

5 Experimental Results

The model was trained using TensorFlow with Adam optimizer and a learning rate of 0.0005. The training process included techniques such as Early Stopping and ReduceLROnPlateau to prevent overfitting and improve learning efficiency. The training spanned 50 epochs, with a batch size of 16 images per batch.

Figures 4 and 5 showcase the Handwriting to Text Converter web application. The interface allows users to upload an image of handwritten text and convert it into digital text instantly.

Performance Metrics

The model's performance was assessed using three key metrics: Character Error Rate (CER), Word Accuracy Rate (WAR), and Inference Speed [24].

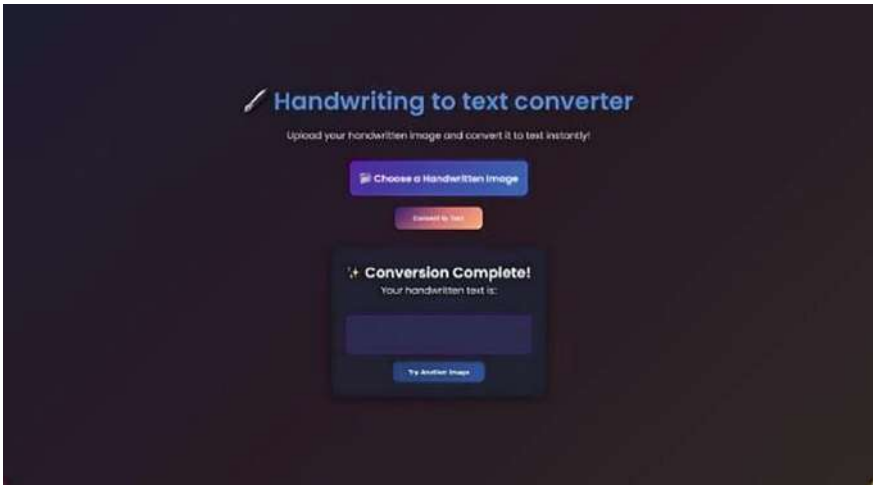


Fig. 4 Screenshot: handwriting to text converter—user interface

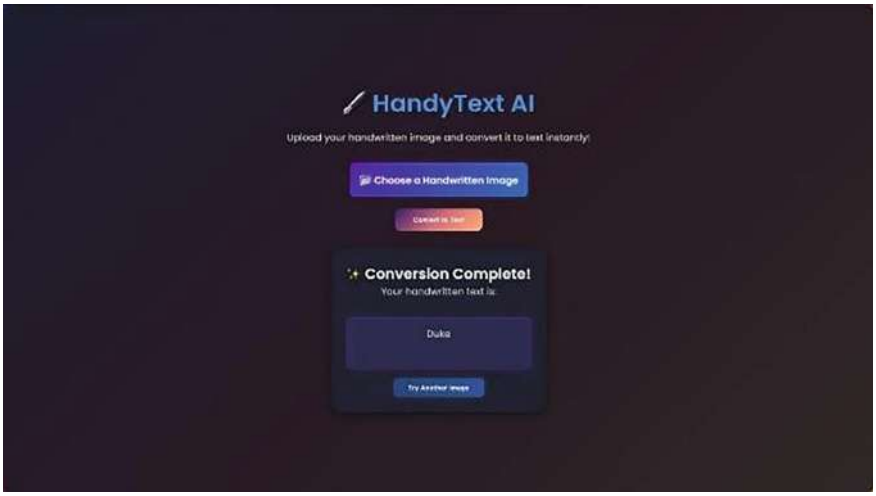


Fig. 5 Screenshot: text recognition

a. CER measures the accuracy of character-level recognition, with a lower CER indicating better performance. This metric helps evaluate how well the model identifies individual characters in handwritten text.

CER gives a proportion of incorrect predicted characters to total number of characters in actual text.

$$CER = \frac{(S + D + I)}{N}$$

where

S is the number of incorrect substitutions, D is the number of missing characters and, I is the number of extra insertions.

b. WAR assesses the percentage of correctly recognized words in the dataset, providing insight into the model's effectiveness at word-level prediction.

$$\text{WAR} = \frac{C}{T} \times 100$$

where

C is the number of correctly recognized words and, T is the total number of words in the actual text.

c. Inference Speed measures the time taken to process a single image and extract text predictions, which is crucial for real-time applications.

Table 1 Performance metrics

Metric	Value (%)
Character error rate (CER)	4.2
Word accuracy rate (WAR)	92.8
Inference speed (FPS)	25 images/s

$$\text{Inference Speed} = \frac{\text{Total Processed Images}}{\textit{Time Taken}}$$

High inference speed is better for real time applications.

Table 1 shows the performance metrics obtained for the proposed system. These values indicate high model performance.

These metrics collectively determine the efficiency and accuracy of the handwriting recognition system.

6 Conclusion and Future Scope

Handwritten Text Recognition (HTR) is essential to postal automation, document digitisation, and other assistive technologies. In this study, a system that can identify handwritten texts by handwriting by using a trained ResNet-based Convolutional Neural Network (CNN) for feature extraction and a network based on Bidirectional Long Short-Term Memory (BiLSTM) for sequence learning is created and implemented. The system's performance is validated by training the model with a variety of handwritten examples from the IAM Dataset. The accuracy of the model's recognition of handwritten characters was observed to be good. The model could propagate through text of any length using the CTC loss function, therefore it didn't require patching or segmentation into smaller sections beforehand. A handwriting recognition system based on the proposed ResNet-BiLSTM can be improved in many ways. The accuracy of recognition may be improved by employing transformer-based structures or attention procedures, especially when working with long or complex sections. Support for cursive and bilingual handwriting would increase the model's usefulness and versatility. Furthermore, utilizing language models to integrate contextual post-processing can greatly improve the identified text's semantic correctness. Lastly, expanding the offline and online handwriting recognition system can have significant real-world implications in the banking, historical document digitalization, and educational sectors.

References

1. Nikitha, A., Geetha, J., JayaLakshmi, D.S.: Handwritten text recognition using deep learning. In: 2020 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT), Bangalore, India, pp. 388–392 (2020)
2. Katoch, S., Rakhra, M., Singh, D.: Recognition of handwritten English character using convolutional neural network. In: 2022 4th International Conference on Artificial Intelligence and Speech Technology (AIST), Delhi, India, pp. 1–6 (2022)
3. Smitha, N., Singh, R.K., Yadav, S.K., Sah, S., Hemanth Kumar, H.S.: An empirical comparison of handwritten character recognition using machine learning. In: 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 1277–1280. IEEE (2021)
4. Singh, P.N., Kiran Babu, T.S.: Recognition of scanned handwritten digits using deep learning. In: 2023 IEEE 3rd Mysore Sub Section International Conference (MysuruCon), pp. 1–6. IEEE (2023)
5. Chi, X., Huang, S., Li, J.: Handwriting recognition based on ResNet-18. In: 2021 2nd International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE), pp. 456–459. IEEE (2021)
6. Ingle, R.R., Fujii, Y., Deselaers, T., Baccash, J., Popat, A.C.: A scalable handwritten text recognition system. In: 2019 International Conference on Document Analysis and Recognition (ICDAR), Sydney, NSW, Australia, pp. 17–24 (2019)
7. RishiKeshan, J., Jeyaseelan, R., KrishnRaj, R., Krishnamoorthy, V.: Handwritten Notes Recognition Using Artificial Intelligence (2023)
8. Ansari, A., Kaur, B., Rakhra, M., Singh, A., Singh, D.: Handwritten text recognition using deep learning algorithms. In: 2022 4th International Conference on Artificial Intelligence and Speech Technology (AIST), Delhi, India, pp. 1–6 (2022)
9. Qi, J., Yang, H., Kong, Z.: Research on handwriting recognition method based on machine learning. In: Third International Symposium on Computer Engineering and Intelligent Communications (ISCEIC 2022), vol. 12462, pp. 624–629. SPIE (2023)
10. John, J.: Support Vector Machine for Handwritten Character Recognition. arXiv preprint arXiv:2109.03081 (2021)
11. Hamid, N., Sjarif, N.N.A.: Handwritten Recognition Using SVM, KNN and Neural Network. https://doi.org/10.48550/arXiv.1702.00723 (2017)
12. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-Based Learning Applied to Document Recognition, Intelligent Signal Processing, pp. 306–351. IEEE Press (2001)
13. Bunke, H., Roth, M., Schukat-Talamazzini. E.G.: Offline Cursive Handwriting Recognition Using Hidden Markov Models
14. Bello, M.I., Adamu, I., Watsilla, H., Umar, K.I.: Multi-layer perceptron network for English character recognition. Am. J. Eng. Res. (AJER) **6**(6), 86–92 (2017)
15. Lekshmy, P.L., et al.: Optical character recognition (OCR) in handwritten characters using convolutional neural networks to assist in exam reader system. In: 2024 2nd International Conference on Advancement in Computation & Computer Technologies (InCACCT), Gharuan, India, pp. 623–627 (2024)
16. Dutta, K., Krishnan, P., Mathew, M., Jawahar, C.V.: Improving CNN-RNN hybrid networks for handwriting recognition. In: 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), pp. 80–85. IEEE (2018)
17. Hemanth, G.R., Jayasree, M., Venii, S.K., Akshaya, P., Saranya, R.: CNN-RNN based handwritten text recognition. ICTACT J. Soft Comput. **12**(1) (2021)
18. Muthukumar, P., Prabhu, A., Karthika, R.A., Eswaramoorthy, K., Rani, V.U., Reshma, V.K.: Handwritten text recognition from image using LSTM integrated with pixel shifting optimization algorithm. In: 2024 International Conference on Advancement in Renewable Energy and Intelligent Systems (AREIS), pp. 1–6. IEEE (2024)
19. Hu, Y., Huber, A., Anumula, J., Liu, S.C.: Overcoming the vanishing gradient problem in plain recurrent networks. arXiv preprint arXiv:1801.06105 (2018)

20. Kizilirmak, F., Yanikoglu, B.: CNN-BiLSTM Model for English Handwriting Recognition: Comprehensive Evaluation on the IAM Dataset (2022). https://doi.org/10.21203/rs.3.rs-2274499/v1
21. Mahadevkar, S., Patil, S., Kotecha, K.: Enhancement of handwritten text recognition using AI-based hybrid approach. MethodsX **12**, 102654 (2024)
22. Kohli, H., Agarwal, J., Kumar, M.: An improved method for text detection using Adam optimization algorithm. Glob. Trans. Proc. **3**(1), 230–234 (2022)
23. Marti, U.-V., Bunke, H.: The IAM-database: an English sentence database for offline handwriting recognition. Int. J. Doc. Anal. Recogn. **5**, 39–46 (2002). https://doi.org/10.1007/s100320200071
24. Garrido-Munoz, C., Rios-Vila, A., Calvo-Zaragoza, J.: Handwritten Text Recognition: A Survey. arXiv preprint arXiv:2502.08417 (2025)

Next-Gen Research Assistance: Autonomous Cognitive Humanoid Lab Assistants for Streamlined Productivity and Safety

Arasa Deekshitha, M. H. Suraj, B. A. Satish, M. H. Rachana, and Y. N. Sharath Kumar

Abstract Innovation is essential in the fast-paced field of contemporary science and study. Lab environments could soon undergo a transformation thanks to the release of the humanoid lab assistant, a ground-breaking combination of Google Assistant and technology. This sophisticated bot can not only move between lab tables on its own, but it can also retrieve data, monitor safety, assist with experiments in real time, and respond to inquiries. Through the AI-driven dialogues it enables, researchers may solve problems together and advance their work to new heights. Because of its integration, Google Assistant is an invaluable tool for researchers looking for information, references, or help. It gives access to a wide range of information resources. This convergence of technologies ushers in a new era of technology-driven research support while also increasing lab productivity and safety. Because of its autonomous mobility, the lab assistant can move quickly around the lab, saving researchers time on manual labour and streamlining their workflow. Additionally, the lab assistant is always available to respond to inquiries and offer researchers advice and help whenever needed. The AI-driven dialogues made possible by are where the actual innovation is found. With the lab assistant, researchers may have lively conversations, brainstorm, and solve problems. This cooperative method pushes the boundaries of discovery and quickens the speed of study. Not only is the lab assistant a tool, it's a scientific partner. The humanoid lab assistant is at the forefront of research assistance technology since it combines features and Google Assistant. Modern laboratory environments greatly benefit from its autonomous mobility, safety monitoring, data

A. Deekshitha (✉) · M. H. Suraj · B. A. Satish · M. H. Rachana · Y. N. Sharath Kumar
Dayananda Sagar College of Engineering, Bengaluru, India
e-mail: deekshitha-eee@dayanandasagar.edu

B. A. Satish
e-mail: satish.eee@dayanandasagar.edu

M. H. Rachana
e-mail: rachana-eee@dayanandasagar.edu

Y. N. Sharath Kumar
e-mail: sharath-eee@dayanandasagar.edu

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_5

retrieval, and collaborative problem-solving capabilities. As we embrace this technological marvel, we usher in a new era of productivity and efficiency in research, thereby reaffirming the role of technology as an essential collaborator in scientific inquiry.

Keywords Lab assistant · Humanoid · Google assistant · Collaborator · Innovation

1 Introduction

Humanoid robots represent an intriguing fusion of state-of-the-art technology with the ancient human urge to build machines that resemble ourselves. These anthropomorphic devices, designed to mirror human physical and cognitive skills, are at the pinnacle of robotics. This comprehensive introduction examines the evolution, guiding principles, applications, and future potential of humanoid robots, emphasizing their significant influence across various domains, including industry, research, healthcare, and entertainment.

The history of humanoid robots can be traced back to ancient myths and fables about automata, which inspired the vision of creating artificial life. However, the practical development of humanoid robots began in the twentieth century when scientific and technological advancements made these concepts feasible. Despite early industrial robotic successes, such as the renowned Unimate in the 1960s, humanoid robots did not gain widespread recognition until the late twentieth century. Their defining characteristic is their human-like form—comprising a head, torso, arms, and legs—which enhances their adaptability to environments and tools designed for humans.

The substantial developments in robotics have led to their broad application in manufacturing, research, medicine, defense, and education. Robots have evolved from simple tools assisting humans to sophisticated machines capable of mimicking human behavior and even replacing human labor in challenging conditions. This transformation not only concerns the physical production of robots but also focuses on the control and optimization of tasks they perform. The study of robot kinematics, localization, mapping, and path planning plays a crucial role in advancing humanoid robotics. Additionally, control systems—comprising actuators, sensors, and processing units—are pivotal in enhancing robot functionality. The experimentation with humanoid robots for civilian use involves programming and control through microcontrollers, with an emphasis on degrees of freedom and movement precision.

Humanoid robot capabilities have been significantly improved by advances in materials science, artificial intelligence, and sensory technology. Flexible and durable materials allow for natural movement, while sophisticated sensors, such as touch and vision sensors, enhance perception. Artificial intelligence, natural language processing, and machine learning enable humanoid robots to understand and respond

to human interactions more effectively. Their applications span numerous industries, showcasing their versatility. In manufacturing, they assist in tasks that demand precision and adaptability, collaborating with humans to improve productivity. Research institutions use humanoid robots to explore human movement, cognition, and social interaction. In healthcare, they aid in patient care and routine medical tasks. Furthermore, they serve as interactive teaching aids in educational settings, enhancing the learning experience. The entertainment industry also integrates humanoid robots, blending fiction with reality. Additionally, the potential for robots to provide companionship and emotional support is under exploration, expanding their societal role.

The impact of humanoid robots extends beyond their immediate functions, influencing cognitive science and human–machine interaction. Their ability to perform embodied experiments allows researchers to test hypotheses on human anatomy and physiology, contributing to advancements in cognitive research. While popular culture has long been fascinated with human-like robots, their real-world implications are now becoming more tangible. Initially considered expensive novelties, commercially available humanoid robots are gradually finding roles in households, industries, government facilities, hazardous environments, and even space missions.

A novel aspect of humanoid robotics research involves the integration of artificial intelligence systems such as OpenAI's ChatGPT. A few-shot learning approach enables humanoid robots to convert natural language instructions into executable behaviors, facilitating adaptive task planning in diverse environments. This involves programmable prompts interacting with execution systems and image recognition software, allowing for real-time decision-making and reduced memory dependence. Studies have demonstrated the effectiveness of this approach in household settings, where robots successfully perform multi-step tasks with minimal user intervention. However, challenges remain in achieving a uniform methodology for task planning, highlighting the ongoing need for refinement and optimization.

Despite remarkable progress, humanoid robots still face significant challenges, including achieving full autonomy, enhancing energy efficiency, and addressing ethical concerns. The complexity of their design necessitates continuous research into their physical structure, sensory capabilities, cognitive abilities, and motor functions. Advances in robotics, AI, and machine learning will be instrumental in shaping the future of humanoid robots, enabling their widespread integration across multiple industries. The potential applications include roles in space exploration, disaster response, and personalized assistance, where they may become indispensable companions in daily life.

The development of autonomous control systems for micro-robots further highlights the increasing complexity of robotic functions. Research in magnetic actuation and Hall effect sensor arrays demonstrates the feasibility of remote control and closed-loop feedback mechanisms, enhancing robotic precision. Such advancements contribute to the broader field of humanoid robotics by improving actuation, detection, and mobility in small-scale robotic systems.

To summarize, humanoid robots represent the human aspiration to create intelligent machines that mimic our form and function. This paper provides a detailed

review of their evolution, design principles, applications, and future prospects as they continue to integrate into human life. As robotics technology advances, humanoid robots have the potential to revolutionize how humans interact with machines, transforming the way we live and work in the robotics era [1–5]. The main aim of this work is designing the multifunctional humanoid assistant capable of autonomous navigation, real-time experiment assistance, contextual data retrieval, safety monitoring, and interactive AI-driven dialogue with researchers by reducing manual labor, fostering collaborative problem-solving and rapid decision-making within the lab. By integrating speech-based AI capabilities with mobile robotics, the assistant can act as an intelligent research partner.

2 Related Works

The field of humanoid robotics and autonomous locomotion has seen rapid advancements in recent years, with research focusing on control mechanisms, perception, human–robot interaction, and applications in real-world scenarios. This literature survey reviews and compares key papers in this domain, covering recent trends, methodologies, and innovations.

Ahirwar et al. [6] provide an overview of advancements in humanoid robot technology, emphasizing improved control algorithms, artificial intelligence (AI) integration, and enhanced sensory perception. Popesku et al. [7] propose an autonomous modular robot, "Active Wheel," designed for dynamic and adaptive movement.

Sartoretti et al. [8] focus on proprioceptive-inertial locomotion for articulated robots, ensuring stable movement without relying on external sensors. Guitron et al. [9] demonstrate magnetic localization-based locomotion in untethered origami robots, showcasing novel mobility strategies in constrained environments.

Meng et al. [10] explore biomechanical evaluations to improve humanoid robot fall control, making comparisons to human falling mechanisms. Kim et al. [11] propose a bipedal walking pattern generator for HUBO, a humanoid robot, allowing stable walking motion.

Similarly, Kaneko et al. [12] discuss modifications in humanoid robot HRP-2Kai for disaster response, highlighting structural reinforcements and improved actuator mechanisms. Yi et al. [13] develop real-time imitation learning techniques for humanoid robots, enabling active stabilization and mimicry of human operators.

Asfour et al. [14] propose a new sensor modality for humanoid robots to detect surface orientation, enhancing their environmental awareness.

Sharma et al. [15] integrate IoT and computer vision in office automation, demonstrating practical applications of autonomous robotic systems. Sikarwar et al. [16] focus on real-time biometric verification using face embeddings, highlighting security applications in humanoid robots.. Zhong et al. [17] present a novel robot-camera calibration technique for constrained environments, improving visual perception accuracy.

The reviewed papers collectively contribute to the evolution of humanoid robotics, addressing locomotion strategies, perception, control, and human–robot interaction [18]. While significant progress has been made, challenges remain in optimizing energy efficiency, real-time decision-making, and adaptability in dynamic environments. Future research should focus on integrating AI-driven learning models and improving real-world deployment capabilities of humanoid robots.

Traditional lab automation systems such as liquid handlers, robotic arms, and sample management systems (e.g., **Tecan**, **Hamilton**, and **Opentrons**) are widely adopted in pharmaceutical and biochemical labs for repetitive, high-throughput tasks. While these systems offer high precision, they are rigid in function and hard-coded for specific tasks, stationary requiring dedicated workspace and lacking adaptive AI or voice-based interfaces [19].

Mobile robots like **Fetch Robotics**, **TurtleBot**, and **Pepper** have been explored for logistics and light interaction in research and healthcare facilities. These robots can navigate physical environments using SLAM (Simultaneous Localization and Mapping) techniques, but they are often used for **material handling**, not scientific collaboration, lack integration with domain-specific AI for experimental procedures [20].

3 Proposed Methodology

Such a comprehensive feature set requires a rigorous development technique to create an AI Cognitive Bot. In order to ensure alignment with these aims, the process starts with defining the needs and goals, as well as the target market and overall objectives. This is followed by the compilation of comprehensive specifications for each feature. The following step involves configuring the hardware and software, which includes putting together the parts that are required, such as displays, microcontrollers, and software frameworks, and installing the libraries and dependencies needed for functions like image recognition and natural language processing.

Subsequently, a secure database is put in place for attendance logs, and face detection and recognition algorithms are integrated into the system for managing attendance. Next is object identification, which makes use of camera feed algorithms and open-source databases to accurately identify items. The creation of personalized chatbots based on user queries improves engagement, while integration with Google Assistant expands the bot's information accessibility. Using OCR techniques to create text extraction modules and defining page numbers for extraction are necessary steps in extracting content from PDFs.

The development of Bluetooth control mechanisms allows bots to understand human inputs for movement, while motor movements are controlled by self-driving motion algorithms and navigation is made possible by camera-based path tracking. Language detection methods are used in addition to APIs to provide language translation capabilities. Text extraction is possible with OCR from camera feed modules, and interfaces that are easy to use are produced by integrating tablets and displays.

Techniques such as YOLOv8 are used to develop personalized picture recognition databases for improved identification accuracy.

The process further includes creating Python quiz applications, designing a sturdy body with moving arms for hardware support and interactivity, and thorough testing and iteration of each feature and the system as a whole. User feedback is continuously gathered to refine design and functionality, ensuring the bot's reliability, adaptability, and proficiency in its intended activities. Through iterative testing, refinement, and continuous improvement, an AI Cognitive Bot with a wide range of capabilities can be efficiently developed to meet diverse needs, benefitting users with its reliability and performance.

As shown in Figs. 1 and 2, an AI cognitive autonomous bot's architecture is well thought out, with each part being essential to the bot's ability to work intelligently and independently. The block diagram offers a visual depiction of these elements and how they word together, shedding light on how sensory information, judgment, learning, and environmental interaction are all seamlessly integrated.

- Sensors (Input Layer): The "Sensors" block, which stands for the various input devices that the bot uses to gather data from its environment, is where the adventure starts. The bot's sensory organs are its touch sensors, microphones, and cameras, which gather data in its raw form for processing.
- Perception Module (Data Processing): The "Perception Module" is directly attached to the Sensors block. This module serves as the brain's first processing unit, processing raw sensory data through the use of computer vision and audio processing techniques. It combines several inputs to create a cohesive picture of the environment the bot is in.
- Intelligent Analysis Decision-Making Module: The "Decision-Making Module" assumes a central role, leveraging the data that has been analyzed. This block contains the bot's intelligence, which makes decisions by evaluating data from the Perception Module. The bot's decision-making skills are aided by rule-based systems and machine learning algorithms.
- Learning and Adaptation (Continuous Improvement): The "Learning and Adaptation" block is located next to the Decision-Making Module. This part, which incorporates techniques like machine learning and adaptive algorithms, is crucial to the bot's development. The bot adjusts to changing conditions, improves decision-making techniques, and gains knowledge from its experiences.
- Control System (Execution of Decisions): The "Control System" block is where the bot's decisions are implemented. This part converts decisions into commands that may be carried out, telling the actuators what to do. The actuators, which are motors and servos, carry out the bot's environmental responses.
- Actuators (Physical Interaction): The "Actuators" block, which is situated after the Control System block, is where the decisions made by the bot are physically expressed. The commands are translated into actions by motors and servos, giving the bot actual interaction with its environment.
- Communication Module (Interaction with External Entities): The bot's capacity to interact with users or other external systems is a critical component of its operation.

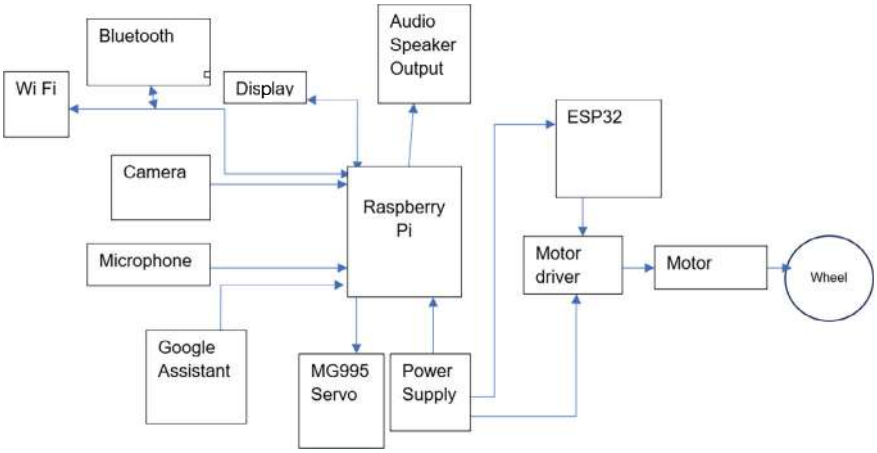


Fig. 1 Block diagram of the proposed work

During this exchange, natural language processing is used by the "Communication Module" to ensure smooth communication. This block improves the bot's usability by ensuring that it can comprehend and produce human language.

- Memory/Storage (Information Retention): The "Memory/Storage" component is connected to several different modules. This part acts as the bot's memory, holding data about its surroundings, prior encounters, and learned material. It enhances the bot's capacity to continue learning and develop its decision-making skills over time.
- Features for Safety and Security (Maintaining Robustness): It is crucial to guarantee the environment and the safety and security of the bot. Positioned strategically, the "Safety and Security" block monitors the entire system. This block creates a strong base for the bot's operations by incorporating features and safeguards against threats and harmful activity.

Circuit diagram of the proposed work is shown in Fig. 2. Carefully crafted to offer users a smooth and engaging experience, the AI Cognitive Autonomous Bot is a ground-breaking combination of hardware and software components. This complex robotic system adheres to a well-organized workflow, where each stage advances the system's overall adaptability and functionality.

The AI Cognitive Autonomous Bot is a flexible and interactive robotic system that seamlessly combines hardware and software components. Each component enhances the overall functionality and user experience of the bot, from the fundamental processes of power initialization and motor control to the complex elements of voice input and auditory feedback during human interaction. The bot's capabilities are enhanced by the addition of visual data, wireless networking, and an interactive display, which positions it as a flexible and futuristic technology with applications ranging from user communication to autonomous navigation.

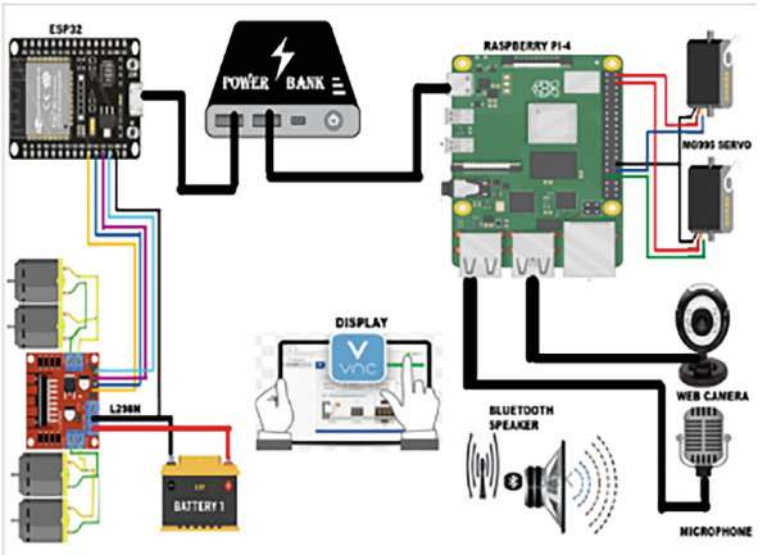


Fig. 2 Circuit diagram of the proposed work

The focus on a dependable power startup guarantees the bot's ability to operate effectively and reliably. Precise locomotion is made possible by the careful management of motors, which allows the bot to adapt to a variety of jobs and situations. The addition of a Raspberry Pi Camera expands the bot's visual capabilities and enables real-time perception and response to its environment. The bot is now more approachable and user-friendly thanks to the addition of voice input and auditory feedback, which elevates human–robot interaction to a new level. The bot's use is increased by the addition of wireless connectivity, which enables data sharing and remote control. Lastly, by offering a tangible interface through which users can communicate with the bot and get immediate feedback, the interactive display raises user engagement. All things considered, the AI Cognitive Autonomous Bot is a technological marvel that exemplifies the potential of fusing cutting-edge software and hardware components with robotics. With applications ranging from interactive user help to autonomous navigation, its versatility, user-friendly interface, and multiple features position it as a versatile tool. The AI Cognitive Autonomous Bot is proof of the promise of intelligent robotics to improve our daily lives and push the limits of what is achievable in human–robot collaboration as technology advances.

4 Results and Discussion

1. Utilizing image recognition to track attendance

When utilizing algorithms to evaluate photos or video streams in order to identify people, image recognition is used for attendance tracking. This technology can be used to automatically record attendance in a variety of situations, including offices, events, and classes, by recognizing faces or other distinctive features as shown in Fig. 3. It improves productivity, lowers errors, and streamlines the attendance process by doing away with the need for manual input. It also provides a quick and non-intrusive means of tracking attendance while upholding privacy rules. This strategy benefits companies and institutions in a variety of industries by automating a previously time-consuming process using advances in artificial intelligence.

2. Recognizing objects with an open source database already in place

To train machine learning models for object recognition, one way to use an open source database is to take advantage of an existing collection of tagged photos as shown in Fig. 4. These databases, like COCO or ImageNet, offer enormous volumes of categorized visual data that are useful for training algorithms to recognize and distinguish between different objects in pictures or videos. The amount of time and resources needed for data gathering and annotation is greatly decreased by using this method. Developers can improve the accuracy and robustness of their models by utilizing an open source database, leading to more efficient applications in domains such as autonomous driving, retail, healthcare, and security. Open source databases encourage cooperation and creativity, which enables programmers to create increasingly sophisticated recognition algorithms.

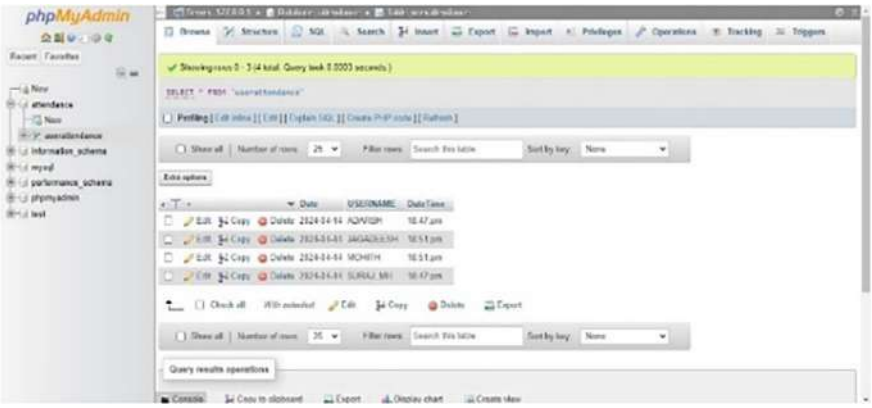


Fig. 3 Attendance record

Fig. 4 Object recognition

3. A Google assistant to access an enormous amount of information

A Google Assistant is a voice-activated digital assistant that accesses a great quantity of information and performs tasks using artificial intelligence. It uses natural language processing to communicate with users, comprehending their inquiries and providing prompt answers. Users can ask it anything, from complicated inquiries and local business information to weather updates and news briefs as shown in Fig. 5. Google Assistant is extremely skilled in information retrieval, scheduling, controlling smart home devices, and much more because it is integrated with Google's vast search database and many services. With its seamless interface for technological engagement and its very user-friendly design, it is accessible on phones, smart speakers, and other connected devices.

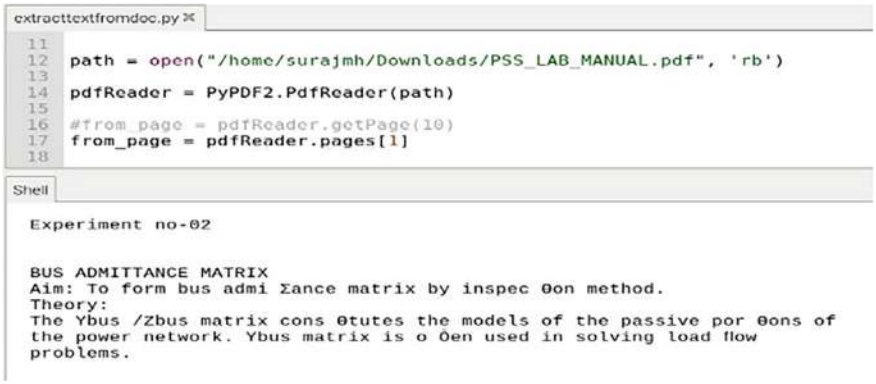


Fig. 5 Google assistant

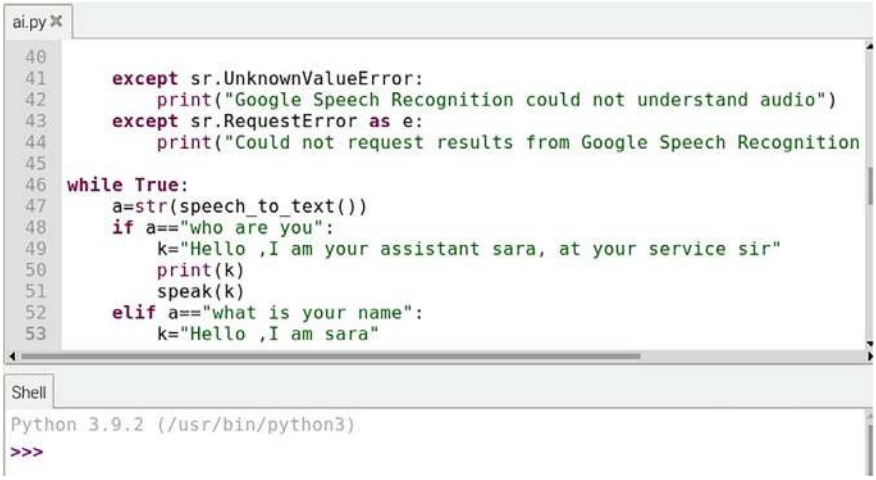


Fig. 6 Custom chatbot

4. A personalized chatbot that can respond to personalized inquiries

A sophisticated artificial intelligence (AI) tool called a personalized chatbot is made to communicate with users using conversational interfaces, including messaging applications or websites as seen in Fig. 6. It is designed to comprehend and react to specific queries depending on context, history, and user preferences. Machine learning and data analytics enable the chatbot to learn from every encounter, gradually refining its responses and suggestions, and this personalization is made possible. These chatbots are capable of doing a wide range of jobs, such as scheduling appointments, offering tailored shopping recommendations, and responding to customer support inquiries. They provide a more effective and entertaining user experience, enhancing the intuitiveness and responsiveness of interactions to individual demands.

5. The automated system that can interpret material from PDFs

Optical Character Recognition (OCR) and Natural Language Processing (NLP) are two cutting-edge technologies that are used by an automated system to understand content from PDF documents to extract and analyse text. Either scanned or digitally produced static PDF documents can be transformed into editable and searchable data using this technique as shown in Fig. 7. The system may process the data further once the text has been retrieved, allowing for features like document summaries, key point identification, and multilingual text translation. By automating data entry and content management processes, these systems greatly reduce manual effort and improve productivity. They are commonly used in the legal, academic, and corporate sectors to optimize workflows.

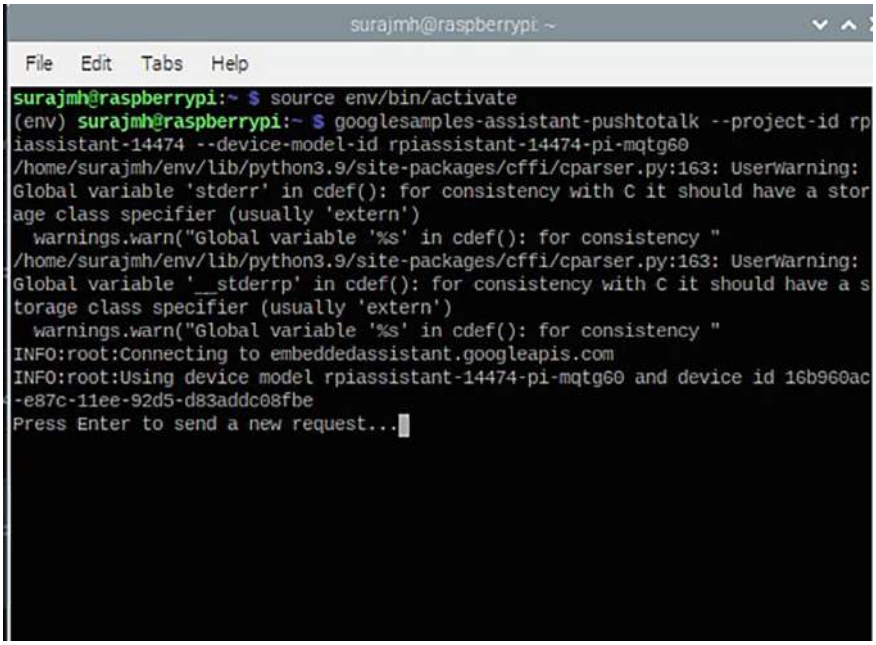


Fig. 7 The bot can read the contents from the document

6. A Bluetooth remote control with button, accelerometer, and joystick modes for operating the robot

For controlling a robot, a Bluetooth remote control with buttons, an accelerometer, and a joystick provides flexible and user-friendly alternatives as shown in Fig. 8. Simple commands like "start," "stop," and "execute specific function" can be performed with the buttons. The robot can imitate the tilt or motion of the remote by using the accelerometer, which provides dynamic control by converting the movements of the remote into commands. For intricate navigation operations, the joystick's accurate, real-time directional control is perfect. Through the use of Bluetooth technology, this multipurpose remote ensures dependable and responsive contact with the robot. In robots for education, entertainment, or industrial applications where flexibility and user-friendliness are critical, such a remote is very helpful.

7. Self-governing movement along a predetermined path by utilizing time delays

A robot or autonomous vehicle that follows a predefined path based on timed instructions is said to be self-governing when it uses time delays. With this system, movement is dictated by predetermined schedules that specify the time of each action (forward, backward, and turning). Rather than relying on sophisticated sensors or real-time data from the environment, the system depends on the precise timing of every movement. This method works best in controlled contexts where barriers are

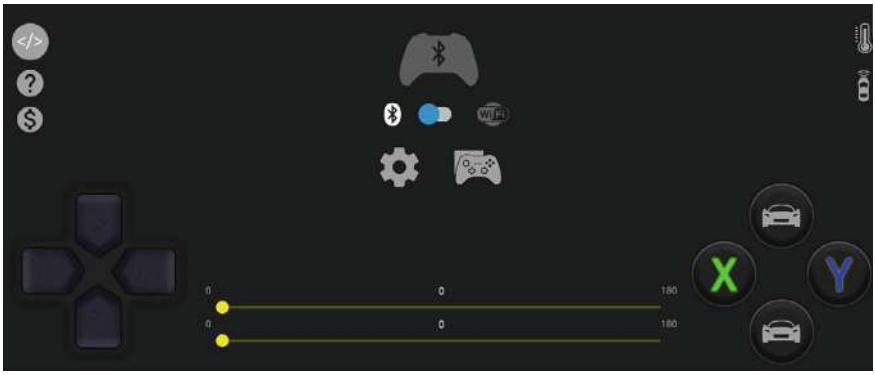


Fig. 8 Bluetooth control

predictable or non existent, despite being relatively easy to execute. It works well in scenarios when there are few external disruptions and the path is constant, such as automated production lines or performance-based activities.

8. Text extraction and OCR utilizing a camera are other capabilities

Optical Character Recognition (OCR) and text extraction with a camera entail taking text images and turning them into editable and searchable digital text. This technique analyzes an image's character forms and lines using sophisticated algorithms to separate the characters from the given backgrounds and other factors. The text is digitized after it has been identified, enabling additional processing including content management, data entry, and translation as shown in Fig. 9. When it comes to digitizing printed documents, automating data entry from forms or receipts, and helping visually impaired individuals read text through audio output, optical character recognition (OCR) is especially helpful. By combining digital and physical information, it improves accessibility and efficiency.

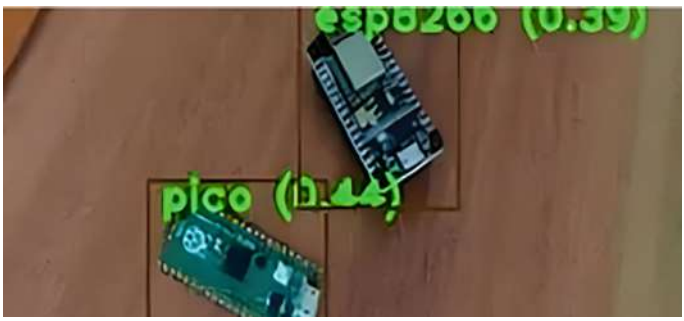


Fig. 9 Text extraction

Fig. 10 Image recognition

9. A personalized photo database for Yolov8 image recognition

A personalized photo database for YOLOv8 image recognition involves compiling a dataset of images tailored to specific recognition tasks. YOLOv8, a deep learning algorithm, excels in object detection and localization within images as shown in Fig. 10. By curating a database with diverse examples relevant to the intended application, such as custom objects or unique contexts, the model can be trained more effectively. This personalized approach enhances the algorithm's accuracy and robustness, enabling it to recognize and classify objects with greater precision in real-world scenarios. Whether for surveillance, autonomous vehicles, or industrial automation, this customized dataset empowers YOLOv8 to perform optimally in specific domains.

10. The Python Quiz app

The Python Quiz application is a dynamic learning resource created to assess and improve users' proficiency with the Python programming language as shown in Fig. 11. Users can choose from a range of questions covering fundamental ideas to complex subjects, and they can get immediate feedback on their level of skill. It provides a dynamic learning environment with features like progress tracking, timed quizzes, and multiple-choice questions. The program is appropriate for both novice and seasoned Python developers, as it may provide resources for more learning and explanations of correct answers. This entertaining and educational tool encourages skill development and mastery by making learning enjoyable and approachable.

Figure 12a, b shows the front and back view of the working model respectively.

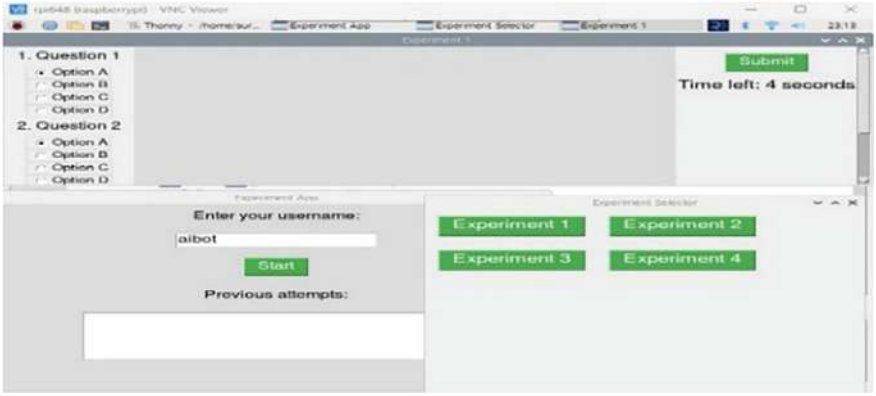
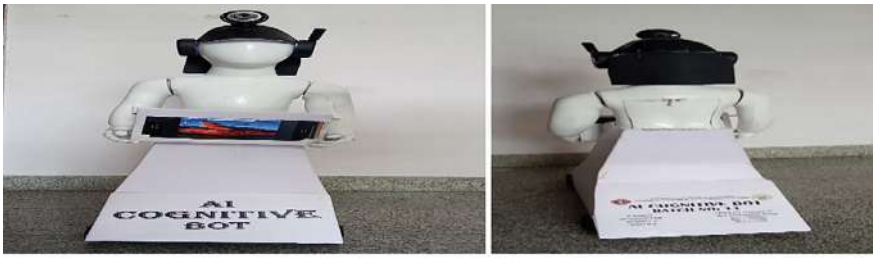


Fig. 11 The quiz app



(a) (b)

Fig. 12 **a** Front view of bot. **b** Back view of bot

The model was tested for the following conditions and multiple cases:

1. Utilizing image recognition to track attendance

Case 1. Proper light

Capturing photos of people in appropriate lighting conditions is necessary when using image recognition for attendance tracking in order to guarantee precise identification. Through picture analysis, the system can identify and log attendance automatically, removing the need for human input and increasing productivity in a variety of settings as shown in Fig. 13.

Case 2. Improper light

The technology can reliably track attendance even in low light thanks to sophisticated image recognition algorithms. The system can adapt to low light conditions by utilizing advanced image processing algorithms, guaranteeing accurate recognition and documentation despite lighting difficulties in the surrounding environment as shown in Fig. 14.

Fig. 13 Proper light

Fig. 14 Improper light

2. Using the Coco database and Yolo v8 algorithm for object recognition

Case 1. Object count

The technology can reliably track attendance even in low light thanks to sophisticated image recognition algorithms. The system can adapt to low light conditions by utilizing advanced image processing algorithms, guaranteeing accurate recognition and documentation despite lighting difficulties in the surrounding environment as shown in Fig. 15.

Case 2. Overlapping objects

Precise item identification is made possible even in situations where objects overlap thanks to the YOLOv8 algorithm's integration with the Coco database as shown in Fig. 16. YOLOv8's strong design makes use of the many object categories and annotations offered by the Coco database to efficiently identify items in congested settings.

Fig. 15 Object count

```
Playing MPEG stream from audio.mp3 ...
MPEG 2.0 layer III, 32 kbit/s, 24000 Hz mono

[0:05] Decoding of audio.mp3 finished.
{'person': 1, 'cell phone': 1, 'clock': 1}
High Performance MPEG 1.0/2.0/2.5 Audio Player for Layer 1, 2, and 3.
Version 0.3.2-1 (2012/03/25). Written and copyrights by Joe Drew,
now maintained by Nanakos Chrysostomos and others.
Uses code from various people. See 'README' for more!
THIS SOFTWARE COMES WITH ABSOLUTELY NO WARRANTY! USE AT YOUR OWN RISK!
tcgetattr(): Inappropriate ioctl for device
```

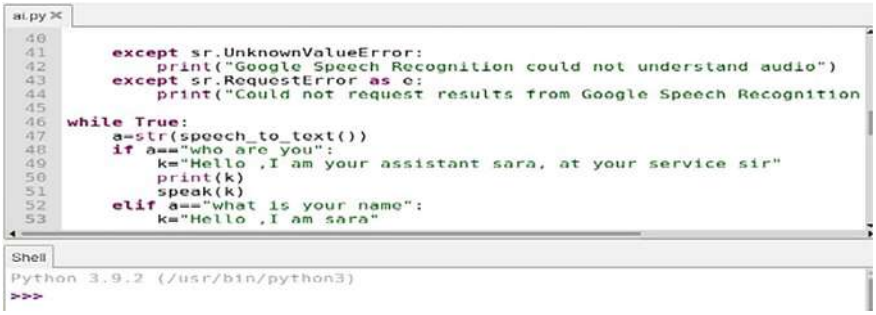


Fig. 16 Overlapping object

This combination makes sure that items are accurately detected and classified, even when they partially obscure one.

3. A Google assistant to access an enormous amount of information

Case 1. In the presence of internet

A Google Assistant that is internet-connected may obtain a vast amount of data from many domains. By utilizing Google's extensive database and robust search features, the assistant can promptly obtain pertinent information, respond to inquiries, deliver updates, and provide support on an array of subjects as shown in Fig. 15. Thanks to its ability to give precise and timely information on demand, this connectivity improves user productivity and convenience (Fig. 17).

Case 2. In the absence of Internet

A locally installed Google Assistant may access a massive offline information reservoir even in the absence of internet access. The Assistant can respond to inquiries based on locally stored data by pre-loading large databases and knowledge bases. This guarantees continuous operation and utility even in offline contexts by enabling users to access information with ease.

4. A personalized chatbot that can respond to personalized inquiries

Case 1. When question is present in the database

A customized chatbot is designed to react to queries that are unique to each user. The chatbot pulls pertinent facts and creates a customized answer when a query matches an entry in its database. The chatbot improves user happiness and interaction with the system by delivering precise and personalized responses by evaluating user input and cross-referencing it with recorded data as shown in Fig. 18.

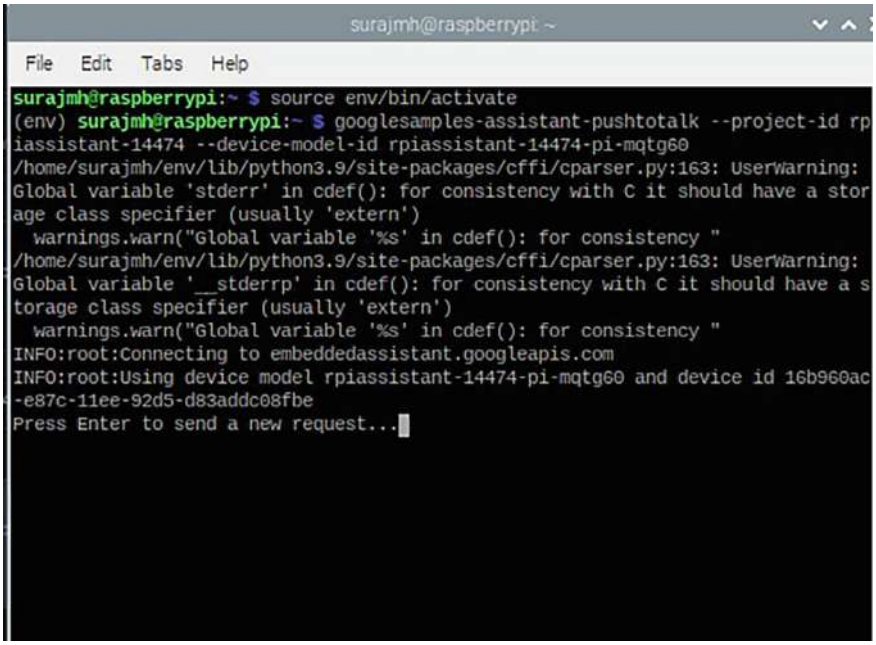


Fig. 17 When internet is present

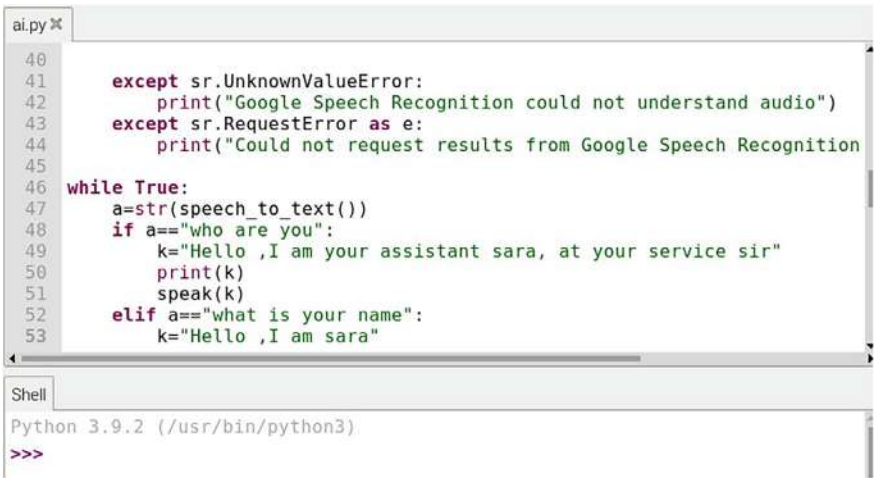


Fig. 18 When question is present

Case 2. When question is absent in the database

Advanced natural language understanding is used by a personalized chatbot to answer specific questions. When a query isn't in its database, the chatbot uses inference and context analysis to provide pertinent answers. It continuously learns from user interactions by utilizing machine learning algorithms, which enhances its capacity to deliver precise and customized answers to a variety of questions, even in unexpected circumstances.

5. The content of the PDF can be read by the bot

Case 1. When text content present in the pdf

The bot can read and comprehend text content from PDF documents because it has PDF text extraction skills. The bot scans the PDF file, extracts text, and transforms it into a machine-readable format using optical character recognition (OCR) technology. This makes it possible for users to easily access the content contained in PDF documents, which makes it easier for the bot to retrieve and use textual data.

Case 2. When text content absent in the pdf

The bot makes information easily accessible by extracting text content from PDF files. When text material is missing or unreadable in a PDF, the bot uses error handling techniques to notify the user and offer substitute alternatives. These solutions could include recommending that the document be rescanned or contacting other sources for help in order to properly fulfill the user's request.

Case 3. When page number wrongly assigned

When the page number is incorrectly assigned, the bot's ability to read PDFs may have trouble reaching the desired content. Nonetheless, the bot can notify consumers of the disparity and ask them to enter a correct page number using error handling techniques. To lessen the effects of improper page assignments, context-aware algorithms might be implemented to assist the bot in locating and retrieving relevant or adjacent content and particular topic.

6. A Bluetooth remote control with button, accelerometer, and joystick modes for operating the robot

Case 1. When bot is in the range

The remote control has multiple operation modes, including button, accelerometer, and joystick modes, while the bot is within Bluetooth range. Users can easily transition between various modes to control the robot in ways that suit their needs and certain tasks. This enhances the user experience and makes it possible for effective navigation and communication with nearby robots by offering intuitive and precise control.

Case 2. When is not in the range

The remote control's functions stop working when the robot is outside of Bluetooth range. When a connection is lost, users are notified via status indicators or notifications on the remote control interface. Users must either troubleshoot connectivity issues or move closer to the robot within the Bluetooth range in order to restore control. By softly returning control upon reconnecting, automatic reconnection techniques can also improve the user experience.

7. The bot can use a camera for OCR

Case 1. When light is present

The bot's camera-based OCR function works best in well-lit environments, taking sharp photos for text extraction. High contrast and sharpness in photos taken are ensured by adequate lighting, which helps the OCR system recognize characters accurately. Consequently, the bot can effectively extract text from photos, allowing visual data to be seamlessly integrated into its processing capabilities for a variety of applications, including data analysis and document digitalization.

Case 2. When light is absent

The bot's camera-based OCR functionality might have trouble correctly extracting text from photos in low-light conditions. In order to address this, the bot has the ability to leverage low-light optimization techniques or add additional light sources, like LED flash, to improve image visibility. As an alternative, users can be asked to improve the lighting in their surroundings in order to increase OCR accuracy.

5 Conclusion and Future Work

Implementing an attendance management system in educational institutions can streamline procedures and minimize manual labor, while a Python quiz app has the potential to evolve into an interactive learning tool. In the corporate sector, attendance management can help track employee work hours, and Google Assistant integration can assist with queries on policies, processes, or FAQs. For customer service, a customized chatbot can provide individualized responses to frequently asked questions, and integrating it with Google Assistant can enhance its ability to handle diverse consumer inquiries. In automation and robotics, improving autonomous locomotion can optimize tasks like warehouse management, where robots transport cargo along pre-planned routes. Additionally, research and development can benefit from a customized image recognition photo database, aiding studies in areas such as medical diagnostics and satellite image-based item detection. For robotics, a Bluetooth-enabled joystick, accelerometer, or buttons can be used for remote control, while an autonomous navigation robot can follow predetermined routes by tracking colored paths or using time-delayed motor control. A multilingual translator can

facilitate communication by supporting multiple language pairs through Google's translation services. Additionally, an OCR text extraction app can digitize handwritten or printed text from images captured by a device's camera. Finally, an interactive educational robot powered by YOLOv8 can feature a 7-inch display or tablet interface to offer educational videos, quizzes, and image recognition capabilities through a personalized photo database.

The main contribution is the The amalgamation of disparate technologies such as robotics, image recognition, database management, and natural language processing into a multifunctional system which has the potential to greatly improve productivity and efficiency in a variety of fields. Through the use of object recognition from open-source databases and picture recognition algorithms for attendance management, the system improves operational efficiency, reduces administrative work, and guarantees accurate record-keeping. The incorporation of Google Assistant enables effortless access to extensive knowledge sources via natural language inquiries, and a customized chatbot provides tailored answers, enhancing user experience and contentment. While Bluetooth remote control and autonomous locomotion capabilities provide intuitive navigation and manoeuvrability for the robot, respectively, they also improve accessibility to textual data through the integration of PDF content extraction and display. Cross-cultural communication is facilitated by the integration of a multilingual translator with OCR capabilities, and item identification accuracy is increased by a customized image recognition database driven by YOLOv8. Furthermore, a sturdy robot body with voice-controlled arms improves physical engagement capabilities, and a quiz Python app promotes interactive learning. All things considered, this all-encompassing system provides flexible functions that could transform entire industries and reshape human–machine connection, ultimately spurring advancement and creativity.

References

1. Somisetti, K., Tripathi, K., Verma, J.K.: Design, implementation, and controlling of a humanoid robot. In: International Conference on Computational Performance Evaluation (ComPE), Shillong, India, pp. 831–836 (2020). https://doi.org/10.1109/ComPE49325.2020.9200020
2. Gupta, P., Tirth, V., Srivastava, R.K.: Futuristic humanoid robots: an overview. In: First International Conference on Industrial and Information Systems, Tirtayasa, Indonesia, pp. 247–254 (2006). https://doi.org/10.1109/ICIIS.2006.365732. In the 19th International Conference on Computer and Information Technology (ICCIT) held in Dhaka, Bangladesh in 2016, pp. 496–500
3. Simul, N.S., Ara, N.M., Islam, M.S.: Presented their work, A support vector machine approach for real time vision based human robot interaction. https://doi.org/10.1109/ICCITECHN.2016.7860248
4. Yokoi, K.: Humanoid robotics. In: 2007 International Conference on Control, Automation and Systems. Seoul, Korea (South), pp. lxxiv–lxxix (2007). https://doi.org/10.1109/ICCAS.2007.4406506
5. Ye, Y., Du, Y.: Improved trust in human–robot collaboration with ChatGPT. IEEE Access **11**, 55748–55741 (2023) https://doi.org/10.1109/ACCESS.2023.3282111

6. Ahirwar, D., Purohit, J., Semwal, V.B., Gawre, S., Rajpurohit, M.: The recent advancements in humanoid robot technology. In: 2022 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), BHOPAL, India, pp. 1–6 (2022). https://doi.org/10.1109/SCEECS54111.2022.9740828
7. Popesku, S., Meister, E., Schlachter, F., Levi, P.: Active wheel—an autonomous modular robot. In: Proceedings of the IEEE 6th Conference on Robotics, Automation, and Mechatronics (RAM), Manila, Philippines, pp. 97–102 (2013). https://doi.org/10.1109/RAM.2013.6758566
8. Sartoretti, G., Ruscelli, F., Nan, J., Feng, Z., Travers, M., Choset, H.: Proprioceptive-inertial autonomous locomotion for articulated robots. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, pp. 3436–3441, https://doi.org/10.1109/ICRA.2018.8460584
9. Guitron, S., Guha, A., Li, S., Rus, D.: Autonomous locomotion of a miniature, untethered origami robot using hall effect sensor-based magnetic localization. In: 17th IEEE International Conference on Robotics and Automation (ICRA), Singapore, pp. 4807–4813 (2017). https://doi.org/10.1109/ICRA.2017.7989560
10. Meng, L., et al.: A falling motion control of humanoid robots based on biomechanical evaluation of falling down of humans. In: 2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids), Seoul, Korea (South), pp. 441–446. https://doi.org/10.1109/HUMANOIDS.2015.7363571
11. Kim, J.-Y., Oh, J.-H., Park, I.-W.: Online biped walking pattern generation for humanoid robot KHR-3(KAIST Humanoid Robot—3: HUBO). In: 6th IEEE-RAS International Conference on Humanoid Robots, pp. 398–403 (2006). https://doi.org/10.1109/ICHR.2006.321303
12. Yi, S.-J., McGill, S.G., Zhang, B.-T., Hong, D., Lee, D.D.: Active stabilization of a humanoid robot for real-time imitation of a human operator. In: 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012), Osaka, Japan, pp. 761–766 (2012). https://doi.org/10.1109/HUMANOIDS.2012.6651605
13. Asfour, T., Weiner, J., Ottenhaus, S., Kaul, L.: The sense of surface orientation—a new sensor modality for humanoid robots. In: 2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids), Cancun, Mexico, pp. 820–825. https://doi.org/10.1109/HUMANOIDS.2016.7803368
14. Sharma, D., Sharma, H., Panchal, D.: In 2020, New Delhi, India hosted the IEEE 17th India Council International Conference (INDICON). Presented "Automatic Office Environment System for Employees Using IoT and Computer Vision," pp. 1–6. https://doi.org/10.1109/INDICON49873.2020.9342455
15. Sikarwar, A., Chandra, H., Ram, I.: Real-time biometric verification and management system using face embeddings. In: IEEE 17th India Council International Conference (INDICON), New Delhi, India, pp. 1–4 (2020). https://doi.org/10.1109/INDICON49873.2020.9342551
16. Ansari, R.J., Karayiannidis, Y.: Task-based role adaptation for human–robot cooperative object handling. IEEE Robot. Automat. Lett. **6**(2), 3592–3598 (2021). https://doi.org/10.1109/LRA.2021.3064498
17. Zhong, F., Li, B., Chen, W., Liu, Y.-H.: Robot–camera calibration in tightly constrained environment using interactive perception. IEEE Trans. Rob. **39**(6), 4952–4970 (2023). https://doi.org/10.1109/TRO.2023.3299533
18. Rafeeq, M., Toha, S.F., Ahmad, S., Razib, M.A.: Locomotion strategies for amphibious robots—a review. IEEE Access **9**, 26323–26342 (2021). https://doi.org/10.1109/ACCESS.2021.3057406
19. Pollard, T.D., et al.: Next-generation laboratory automation: advances and challenges. Nat. Rev. Methods Primers **1**(1), 3 (2021)
20. Lou, Z., et al.: SLAM-assisted mobile robotic platforms in dynamic environments: a review. Robot. Auton. Syst. **164**, 104340 (2023)

Cross Model Communication Sign Language to Text and Speech to Sign Language Using Inception V5

L. Priya and B. Chandrasekar

Abstract Cross model communication, a bidirectional sign language translation system that bridges communication between American Sign Language (ASL) users and non-signers. The system integrates two core modules: (1) a sign-to-text translation pipeline using an enhanced Inception V5 model, and (2) a speech-to-sign translation mechanism utilizing Whisper AI for speech recognition and animated ASL output generation. The Inception V5 model classifies ASL gestures with 90.2% accuracy and an inference time of 50 ms, while the speech-to-ASL module transcribes spoken input and maps it to ASL representations with 94% accuracy. A key contribution of this work lies in its modular architecture, which supports dynamic gesture recognition, robust speech transcription under noisy conditions, and multimodal translation output, including static letter signs and animated word signs. Extensive experimentation confirms the system's reliability across varied lighting conditions, hand orientations, and speech accents. This solution has potential for scalable deployment in educational, social, and assistive contexts. Future enhancements include expanding vocabulary, real-time animated synthesis, and contextual gesture interpretation.

Keywords Sign language translation · Inception V5 · Whisper AI · Deep learning · Real-time gesture recognition · Speech-to-ASL · Accessibility · Deaf and hard of hearing · Communication technology

L. Priya · B. Chandrasekar (✉)
Department of Information Technology, Rajalakshmi Engineering College, Chennai, India
e-mail: chandrasekar291099@gmail.com

L. Priya
e-mail: Priya.l@rajalakshmi.edu.in

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*,
Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_6

1 Introduction

Effective communication remains a fundamental challenge for the Deaf and Hard of Hearing (DHH) community, especially when interacting with non-signers unfamiliar with American Sign Language (ASL). Despite the ubiquity of ASL among DHH individuals, mainstream society often lacks the resources or fluency to bridge this linguistic divide, limiting access to education, employment, and social engagement. To address this persistent gap, we propose a unified, AI-driven, real-time sign language translation system capable of bidirectional communication between ASL users and non-signers.

The proposed system is unique in its dual functionality. First, it translates ASL gestures into readable text using an improved version of the Inception V5 convolutional neural network, optimized for gesture recognition with high speed and accuracy. Second, it enables speech-to-sign conversion by transcribing spoken input using Whisper AI—a transformer-based speech recognition model known for its robustness to accents and background noise—and mapping it to ASL via either static letter images or animated sign word representations.

Unlike prior works that focus solely on one-directional translation or require controlled environments, this system is designed for dynamic, real-world interactions. Its modular structure allows seamless operation across platforms, adaptability to user interaction, and integration of multiple input–output formats. The novelty lies in the cross-model communication strategy that leverages both vision and speech modalities, ensuring accessibility and inclusivity for diverse communication scenarios (Fig. 1).

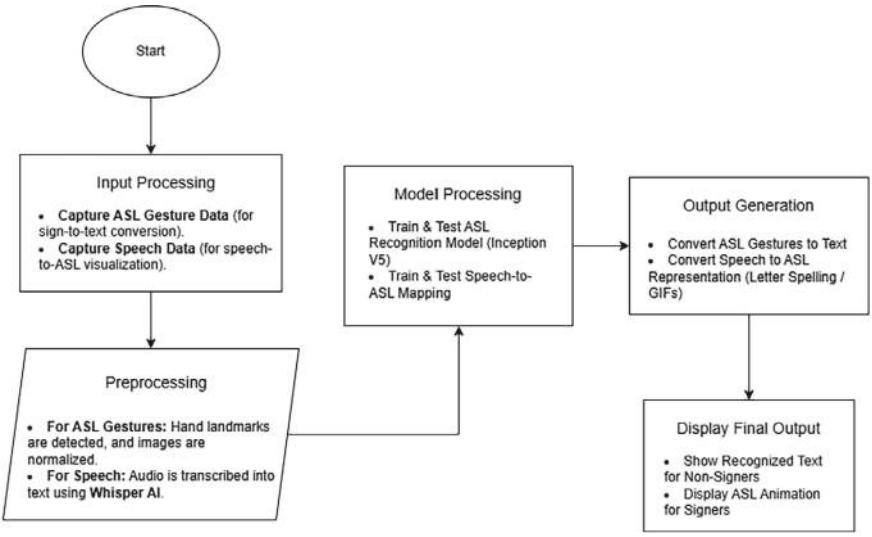


Fig. 1 Conceptual overview

High accuracy and robustness of the system is achieved through advanced preprocessing techniques, data augmentation as well as optimized deep learning architectures. To augment generalization, the gesture recognition model is trained with real world ASL datasets with variations in lighting as well as the hand orientations, and variability in the signing style. Likewise, the speech recognition module utilizes Whisper AI which can process a multitude of various speaking, dialects and environment noise to achieve high precision of transcription prior to ASL mapping. The system is multi–modal, that is, it supports both input types (gestures and speech) and dynamically chooses the output as either text or ASL representations depending upon user interaction.

The system utilizes two advanced AI models: Inception V5 and Whisper AI to allow users to interact seamlessly between a sign language user and a non-signer. ASL gestures are recognized by Inception V5 using computer vision techniques, hand landmarks are detected, gestures are classified and converted into readable text. Whisper AI extracts speech as speech input, transcribes the speaker's words into text, and maps the text to ASL representation through spelling using letters or ASL through GIF animations. In Fig. 2, this dual model approach allows for a full real time translation system between signers and non-signers.

This system is unlike the already existing sign language recognition tools that are primarily text-based translation and adopts a new multi directional sign language recognition that integrates gesture recognition and AI powered speech processing. Dynamic ASL spellings and animated gestures offer a natural, real time conversation experience, as a result of incorporation into the platform. Also, the real-time adaptability of the system with its ability to integrate seamlessly, along with high accuracy guarantees its scalability for applications in educational settings, workplaces, and as a part of assistive technologies, thus improving accessibility for both ASL users as well as non-signers.

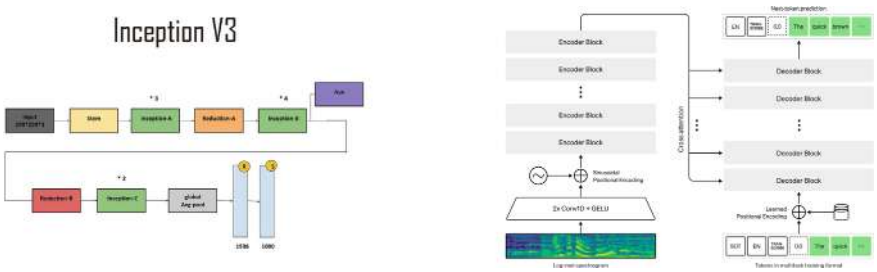


Fig. 2 AI systems for sign language communication

2 Literature Survey

Recent developments in artificial intelligence have led to notable advancements in sign language translation, leveraging deep learning techniques such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and transformer-based architectures. These efforts aim to bridge communication barriers between Deaf and Hard of Hearing (DHH) individuals and non-signers. However, many existing systems remain constrained by limitations in real-time responsiveness, dynamic gesture recognition, adaptability to diverse environments, and support for bidirectional communication.

Deshmukh et al. [1] have developed a deep learning real time Speech-to-Sign Language converter. It converts the spoken words to sign representations constituting the help to the Deaf and Hard of Hearing (DHH) community. Speech is mapped well to sign visuals, but difficulty arises where sentence structures and the meaning associated in context become involved. For future work, natural language processing (NLP) can be incorporated to help interpretation.

Lalitha et al. [2], they introduced a speech to sign language interpreters based on AI recognition models. Spoken language is automatically processed, converted into text and mapped to the corresponding ASL signs. It is effective for common words, but is not adaptable to regional sign variations. The focus could be to expand the gesture vocabulary and real–time feedback could be added.

Sparsha et al. [3] proposed a system, which recognizes gestures using CNN and converts sign language to speech. It recognizes gestures in the hand and translates them in spoken language. The model is accurate, but it is ineffectual in the presence of background noise and varying orientations of subjects' hands. Specifically, its real-world usability can be improved by improving noise filtering and multi angle gesture detection.

Hrithik et al. [4] developed machine learning assisted sign to speech gloves. Attaching sensors to the palm of the hand and wrist, the system can capture hand movements and its output is audio. It is effective for controlled environments as long as it restricts gesture variability and sacrifices real time speed. Better performance could be realized with sensors which have better precision and response time.

Shashidhar et al. [5] have designed an Indian Sign Language (ISL) to speech conversion model using CNNs. The gestures created by ISL are classified by the model and spoken output is generated. It promises a little, and it does have difficulty recognizing fast signing motions and intricate facial expressions. In future work, LSTMs that are temporal models could be integrated to be able to handle dynamic gestures better.

Kowsigan et al. [6] A speech to sign language system with live gesture recognition. The model creates a communication that is more accessible by translating spoken words into sign animations. Yet the system is having problems with homonyms and words that are ambiguous. Improved contextual NLP and gesture animation quality is able to improve effectiveness.

Peguda et al. [7] introduced A speech to sign the translation system for Indian languages. Speech is processed, turned into text, and mapped to ISL gestures by it. It is effective in translating isolated words, but it is not fluent enough in a sentence translation. Continuous gesture generation may be supported in future upgrades to provide for smooth communication.

Om Kumar et al. [8] A real time gesture detection and conversion system with sign to text and speech to sign translation is developed. However, the dual mode approach is better in accessibility but poor in terms of real time speed and accuracy under variable lighting. Preprocessing technique can be optimized along with the real-time inference speed to refine the performance.

Rai et al. [9] A speech to sign language translators using deep learning is developed and its Speech features are extracted by the model, then text and a set of corresponding sign representations are mapped from those features. Specifically, it works well in structured sentences but cannot deal with spoken words. Contextual awareness may be considered for future improvements of the phrase mapping.

Kowsigan et al. [6] A sign language conversion model based on combining speech recognition along with the live gesture detection was proposed. The speech is translated as sign representations in order to improve accessibility. But it has trouble with real time synchronization of speech input to a sign output. This issue can be addressed through enhancing the response time and supporting multimodal integration.

Nurfita Sari et al. [10] A smart glove based two-way sign language translation system with artificial neural network was introduced. It is able to recognize hand movements but has the issues to recognize nonstandard gestures. Improving the system robustness can be obtained by refining sensor calibration and creating more gesture datasets.

Gunvantray et al. [11] A CNN based model for sign to text translation was developed as an application detects static and dynamic gestures and converts them to readable text. However, it obtains high accuracy with the common signs and is not good with the overlapping gestures and movement transitions. Such temporal tracking would have to be integrated into future work to provide smoother translations.

Thong et al. [12] A vision-based sign language to text translation system through computer vision was introduced. It takes the hand landmarks detected by the model and converts them into text. Even though effective for isolated signs, it does not support continuous signing. Improving fluency would involve providing a stronger boost to sequence learning and if possible, incorporating a language model.

Mannepula et al. [13] A review of Sign language-to-emotion specific text translation method where their work shows that emotional context is important for sign communication. Currently, existing models cannot capture such subtle facial expressions and emotions. Further research regarding the fusion of multimodal features would be attempted for superior sentiment recognition.

Seviappan et al. [14] proposed an RNN-LSTM based sign-to-text conversion system. Starting with sequential gestures, they are represented as text in the output. It is effective when applied on structured sentences but does not work well on rapid hand movements or occlusions. Improved transformer dependent frameworks could make them more adaptable in actual time.

Existing sign language translation systems face multiple limitations, primarily in real-time accuracy, contextual understanding, and adaptability to environmental variations. Speech-to-sign systems often struggle with sentence structures, homonyms, and regional sign variations, making translation inconsistent. Additionally, some models are ineffective in dynamic scenarios, struggling with fast signing motions, overlapping gestures, and synchronization of speech input to sign output. Real-time responsiveness is another challenge, with delays in processing speed affecting user experience. In summary, while prior systems have addressed isolated aspects of gesture or speech translation, our work distinguishes itself through a real-time, bidirectional architecture, robust recognition under natural conditions, and scalable deployment potential across educational, assistive, and social settings.

3 Methodology

The method proposed for developing a bidirectional sign language translation system which allows for communication between the sign language user and non-signers is outlined. This system has two core functionalities: real-time ASL gesture recognition (ASL to text) and speech to ASL visualization (speech to ASL representation). The methodology combines preprocessing of the data with the introduction of deep learning models, namely Inception V5 for gesture recognition and Whisper AI for speech transcription, and enhances system architecture. The system targets as high accuracy and efficiency as possible, suitable for different communication scenarios.

3.1 Data Collection and Preparation

This project's dataset consists of a multitude of American Sign Language (ASL) gestures which contain the entire alphabet and commonly used words. To enhance generalization, the gesture dataset is collected from publicly available ASL repositories and real time recorded gestures are added. To guarantee precise training, they have used a label for each image regarding its corresponding character or word. To achieve robustness, data augmentation techniques of rotation, zoom, flip and variable brightness are applied to mimic the real-world conditions and signing variation.

In order to convert speech to ASL, we use Whisper AI, an advanced speech recognition model trained on such a multitude of audio datasets, that allows us to accommodate any accent, speech speed or level of background noise. To be robust to ultimate transcription, the audio data collected are diverse speech samples. It maps preprocessed text output from the transcription to ASL static letter images or animated ASL word signs using an organized resource dictionary which allows a smooth transition from speech to ASL.

3.2 Workflow Design

It works with the proposed system's workflow that includes a series of stages handling efficiently ASL-to-text as well as speech-to-ASL translation. The procedure begins with real-time gesture recognition by a camera which captures ASL hand gestures and classifies them by virtue of InceptionV5. When an ASL sign that is a valid one is detected, it is then converted into text and displayed on the screen. In the absence of any valid sign the system keeps on processing the new frames in real time.

In parallel, the speech-to-ASL pipeline allows the users to choose between uploading an audio file and speaking into a microphone in real time. Then, this speech is transcribed to text by the Whisper AI model, which transmits it to an ASL representational processor which maps the ASL representations for the corresponding text. An ASL word GIF is shown if there is a full one, or, if not, the words are spelled out with static ASL letter images if possible. The real time feedback includes audio playback confirmation, progress indicators and error handling to name a few (Fig. 3).

3.3 Model Architecture and Enhancements

For the recognition of ASL gestures the primary deep-learning architecture used is the Inception V5 model. Also, its multi path convolutional layers serve for efficient feature extraction of the feature map that learns to represent hand gestures, hand orientation, and different types of signing variations. To fix the generalization problem, dense, batch normalization and dropout layers are stacked on top of the CNN part of the model to prevent overfitting. With these modifications, the system will operate equivalently under different environmental conditions and hand movements.

Whisper AI is used as a speech recognition model for speech-to-ASL conversion, where its pre-trained transformer-based architecture allows speech of any kind (any accent, any kind of noise interference) to be processed. The resulting text string output is processed using a string-matching algorithm (difflib) to find the matching ASL word representations from the system's resource library of ASL words. These models together accomplish a realistic real time bidirectional communication for ASL users and non-signers.

The Inception V5 architecture was selected for ASL gesture recognition due to its proven performance in extracting fine-grained visual features across multiple scales. Unlike standard CNNs, Inception networks utilize a multi-path convolutional strategy, which allows simultaneous processing of spatial patterns through different kernel sizes. This characteristic is especially advantageous for recognizing complex and variably oriented hand gestures. Inception V5, in particular, introduces factorized convolutions, auxiliary classifiers, and batch normalization layers that significantly enhance computational efficiency while reducing the risk of overfitting.

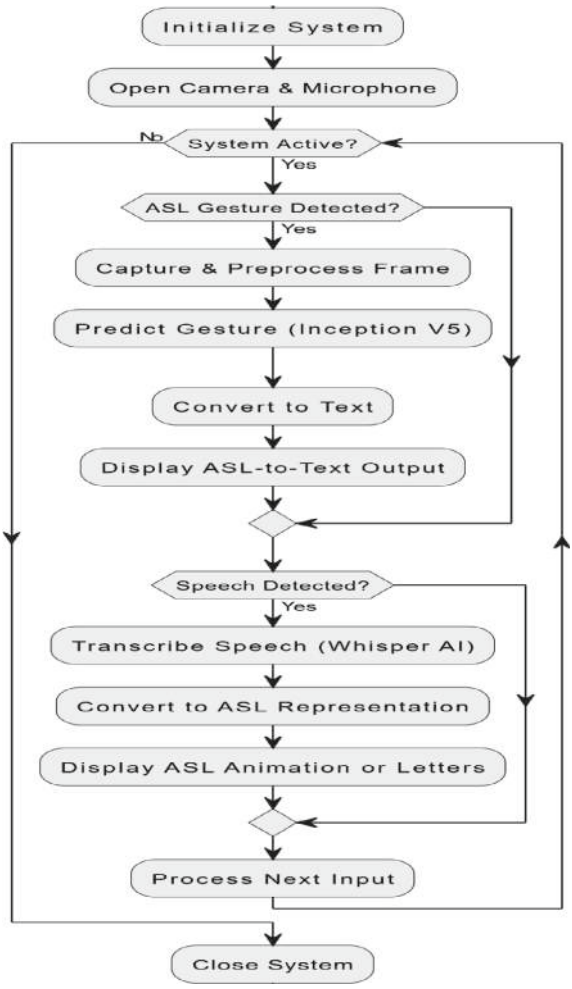


Fig. 3 Workflow diagram

In the context of ASL gesture recognition, where hand shape, orientation, and motion intricacies must be captured under diverse lighting and background conditions, Inception V5 offers superior accuracy with relatively low inference time. Its ability to model hierarchical spatial dependencies makes it well-suited for identifying subtle differences between similar gestures. Furthermore, its lightweight structure ensures real-time classification performance when deployed on consumer-grade hardware, making it practical for live ASL-to-text translation applications.

3.3.1 Training Strategy

The model training process is engineered in such a way that the training process is structured enabling high accuracy and adaptability. To train the ASL gesture recognition model, a categorical cross entropy loss function and the Adam optimizer are used, and several real—world ASL datasets and augmented images are employed. ModelCheckpoint is used to keep track of performance and save the best model as determined by validation loss. Classification report and confusion matrix is regularly used to evaluate and find possible improvements in the recognition accuracy.

Training is optimized for Whisper AI speech-to-text models to fine tune the speech-to-text transcription accuracy. All these steps are hooked by an efficient text normalization and automatic word segmentation system in order to map pronounced words quickly into ASL representations. Fuzzy matching algorithms aid to improve the accuracy of speech to ASL conversion so that there is no hindrance in seamlessly converting speech input to ASL visual representation.

3.3.2 System Architecture

The overall system architecture is designed with a modular approach, allowing both ASL-to-text and speech-to-ASL processes to function independently or together. The system consists of the following core components:

1. Camera Module—Captures ASL gestures in real-time.
2. Preprocessing Module—Resizes, normalizes, and prepares images for model input.
3. Gesture Recognition Module—Uses Inception V5 to classify ASL gestures.
4. Speech Recognition Module—Uses Whisper AI to transcribe spoken input.
5. Text Processing Module—Normalizes and segments transcribed text.
6. ASL Resource Library—Stores static letter images and animated word GIFs.
7. Display/Output Module – Presents recognized ASL signs as text, letters, or GIFs (Fig. 4).

The system is optimized for real time performance and supports being accessed through desktop applications or mobile interfaces. This is modular architecture that ensures scalability for the purpose of having future enhancements such as extended ASL vocabulary, multilingual support, and real time sign language animation synthesis. This is a tremendous step forward towards accessible communication between ASL users and non-signers using state of the art deep learning techniques coupled with real time processing capabilities.

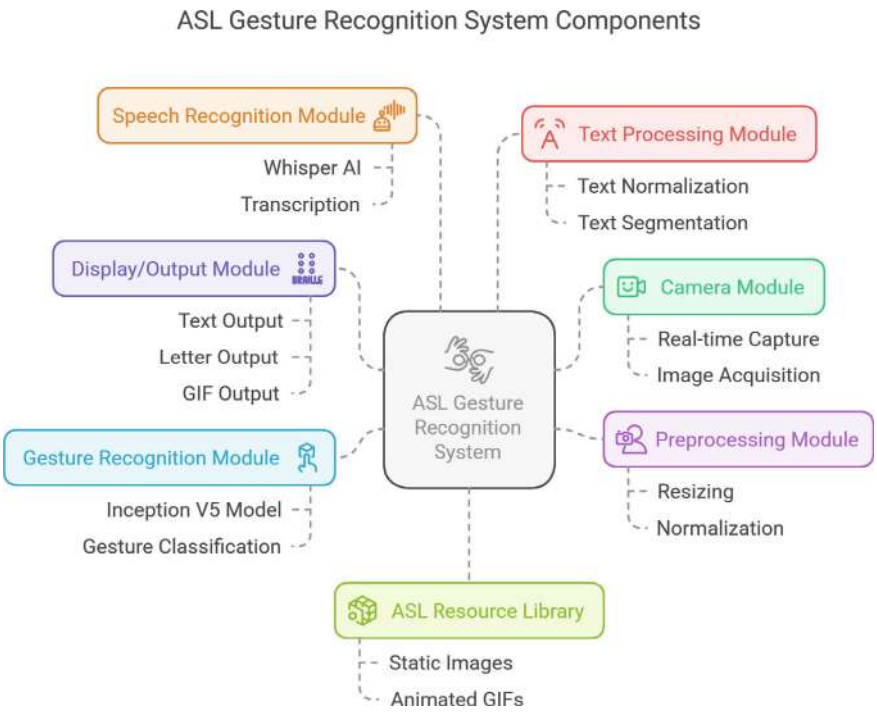


Fig. 4 System architecture

4 Results and Discussion

The real-time performance of the bidirectional translation system is presented in this section and illustrates its capability to generate text and speech from ASL gestures and ASL representation from text and speech. Deep learning models are used to efficiently capture, process and translate input of sign language users into understandable messages for non-signers through the system.

4.1 Sign Language to Text Translation

A webcam is used to capture ASL gestures, landmarks are detected on the hands, and they are classified using Inception V5. Writing one letter at a time, the users sign, as the recognized text is updated around the text dynamically, so that complete words are eventually formed. The system can be seen assembling the word “HAT” in real time (Figs. 5, 6 and 7). The feedback is given in real time and you can adjust the gesture if needed for proper recognition.

Fig. 5 Hand gesture recognized as the letter "H" and "A" in ASL

Fig. 6 Hand gesture recognized as the letter "H" and "A" in ASL

Fig. 7 Completes the word by recognizing the letter "T," displaying "HAT."

4.2 Speech to Sign Language Translation

Whisper AI is used in speech-to-sign modules for transcribed spoken words to ASL representations. The system allows voice input and the user can use prerecorded audio files. The spoken phrase "GOOD NIGHT" is transcribed in Fig. 8, converted

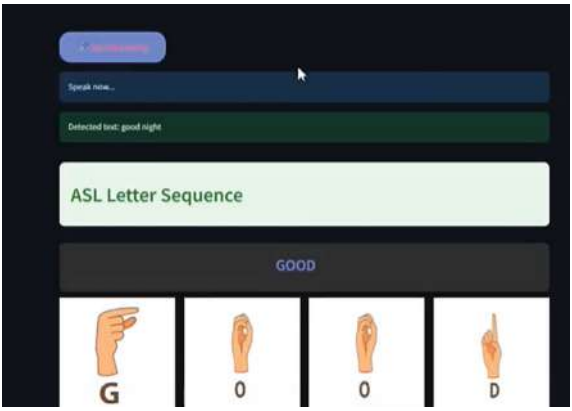


Fig. 8 ASL letter sequence for "GOOD" and NIGHT in speech-to-ASL translation

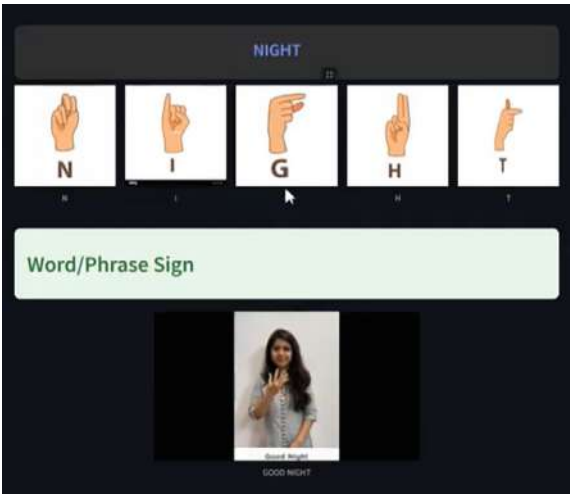


Fig. 9 ASL letter sequence for "GOOD" and NIGHT in speech-to-ASL translation

into ASL letters and displayed as an animated ASL sign in Fig. 9. Transcription is integrated with visual representation to be accessible to ASL users.

4.3 Model Performance Evaluation

The bidirectional sign language translation system was evaluated for performance in terms of accuracy, precision, recall, F1-score and inference time. It includes the evaluation of Sign-to-Text and Speech-to-Sign components independently of each other in terms of their reliability, and efficiency. Real time hand gesture inputs were tested on the Sign-to-text model which was powered by Inception V5. The individual ASL letters were successfully classified by the system with a high degree of accuracy

Table 1 Sign-to-text model performance metrics

Metric	Value (%)
Precision	91.7
Recall	90.5
F1-score	91.1
Accuracy	90.2
Inference time (ms)	50

Table 2 Speech-to-sign model performance metrics

Metric	Value (%)
Accuracy	94.0
Macro avg. precision	96.0
Macro avg. recall	94.0
Macro avg. F1-score	94.0

and this resulted in a smooth, reliable conversion of hand gestures into readable text. The list of key performance metrics is given by Table 1.

Strong generalization across different positions of hands and lighting conditions was found in the model. The latency of ~ 50 ms reflects that the system is fast enough for real time use. The Speech-to-Sign model experienced high accuracy in terms of reading spoken words and signifying them as ASL representations, using Whisper AI for speech recognition. Table 1 shows the total performance of the model (Table 2).

The Speech to Sign model demonstrated excellent performance in processing diverse patterns of speech including difference in pronunciation, and noise in the background with high accuracy. The robustness and macro mean of the precision and recall ensure classification performance on all the ASL letters and hence, reliable visualization of the sign language given by a verbal input.

4.4 Model Accuracy and Loss Analysis

The training and validation accuracy and loss curves of both the Sign-to-Text and Speech-to-Sign models are given in Figs. 10 and 11 respectively. Loss curves also show that the model converges well and accuracy curves show steady improvement in recognition capabilities. We observe minor fluctuations in validation accuracy because of the differences in hand position when performing ASL gestures and the fact that changes in speech affect the transcription quality.

The results of the evaluation indicate that both models are highly accurate and efficient, and are capable of real-time bidirectional communication between ASL users and non-signers. The system can be further optimized through additions to the dataset and real-time error correction in order to enhance the system's robustness in practical deployments.

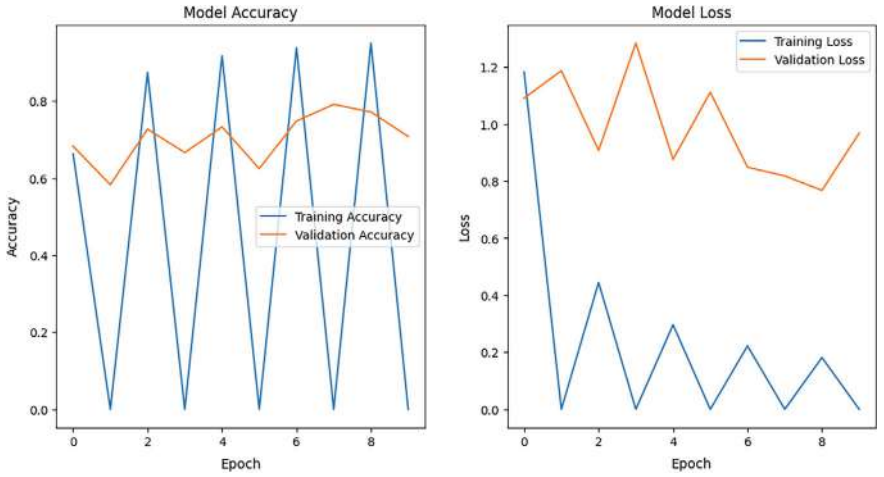


Fig. 10 Model accuracy and loss curves for sign-to-text translation

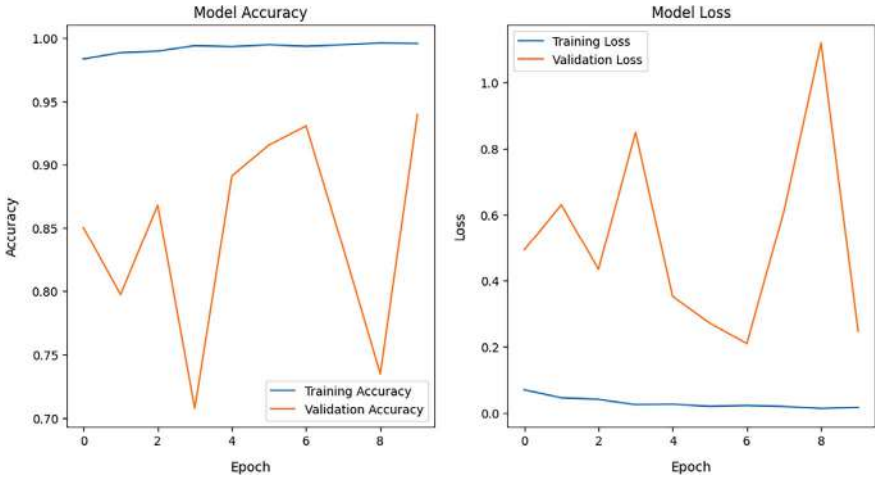


Fig. 11 Model accuracy and loss curves for speech-to-sign translation

The bidirectional sign language translation system allows the Deaf and Hard of Hearing (DHH) to communicate fluidly with the non-signers (those who do not use sign language). With an accuracy of 90.2% on the Sign-to-Text module, provided by Inception V5, the ASL hand gestures can be detected and classified as text with high precision (91.7%) and low inference time (50 ms). This guarantees ASL user's capability to communicate with the non-signers in real time, with the least delay. It was demonstrated that the system could work well in real world scenarios under different lighting, hand orientations and signing styles.

Likewise, the Speech-to-Sign module tracks a speaker's utterance and transcribes the spoken language into ASL representations at 94% accuracy, rendering it quite effective for speech communication. Word transcriptions are accurately mapped to either ASL letter spellings or predetermined sign animations via the system, resulting in the proper representation of what is spoken. There are slight variations in validation accuracy as random people utter speech in different pronunciations and there is interference noise in the background however the system remains stable and responsive processing speech input dynamically and generating precise ASL outputs.

The multi modal integration of this system is one of its key strengths as it allows it to work Bi- directionally between ASL users and non-sign users. This approach works unlike conventional translation tools which do translation as unidirectional translation (either sign to text or speech to text) but this approach is based on deep learning recognition and speech processing for real time and adaptive communication. With additional optimizations through modified preprocessing, data augmentation, and model improvements, the system is further improved in performance and is therefore a modular structure which can scale for educational, commercial and assistive applications. Further, the system's capability of generalizing across different conditions and eliminating the problem of misclassifying repercussions usually found in conventional sign language recognition models primarily due to the variability in hand shape offers good alternative sign language recognition models. Although the system reaches high accuracy and can operate online, challenges remain. Small fluctuations in validation accuracy occur for both modules, with the former suggesting an improvement of gesture detection under occlusions, the latter speech processing in noisy condition, and the latter continuous ASL gesture recognition for fluid ASL to English translation. Further enhancements in the future may entail the widening of the ASL vocabulary, as well as adding in the ability for recognizing gesture based contextual understanding, and finally, the inclusion of animated sign synthesis in real time.

With this research, a new bidirectional communication system is presented that consists of advanced deep learning and AI enabled a transcription system to convey spoken language to visual languages and vice versa. This system is capable of achieving real time interaction and even by virtue of this it provides the base for the inclusive and scalable sign language translation solutions that will pave the way toward the universal accessibility of human communication.

5 Conclusion

This research introduces a comprehensive, real-time, bidirectional communication system that facilitates seamless interaction between ASL users and non-signers. By integrating Inception V5 for accurate ASL gesture recognition and Whisper AI for robust speech transcription, the system effectively bridges the communication gap in dynamic environments. Experimental evaluation demonstrates high accuracy,

precision, and real-time responsiveness, validating its applicability for educational, assistive, and public use cases.

The novelty of the proposed system is reflected in its dual-model synergy, multimodal input handling, and adaptable output generation using both static and animated ASL representations. Furthermore, the modular system architecture supports future scalability and integration with broader language resources. With ongoing enhancements, including animated sign synthesis, contextual gesture interpretation, and multilingual support, the platform has the potential to revolutionize inclusive communication tools for the DHH community and beyond.

References

1. Deshmukh, A., Machindar, A., Lale, S., Kasambe, P.: Enhancing communication for the hearing impaired: a real-time speech to sign language converter. In: 2024 27th International Symposium on Wireless Personal Multimedia Communications (WPMC), Greater Noida, India, pp. 1–5 (2024). https://doi.org/10.1109/WPMC63271.2024.10863135
2. Lalitha, S., Adavi, V.: Beyond words: speech to sign language interpreter. In: 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kamand, India, pp. 1–6 (2024). https://doi.org/10.1109/ICCCNT61001.2024.10725421
3. Sparsha, U., Priyanka, M., Mukthashree, S., Kiran, K.N.: System for sign language to speech conversion. In: 2024 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE), Bangalore, India, pp. 1–4, (2024). https://doi.org/10.1109/IITCEE59897.2024.10467410
4. Hrithik, T.H., Rhethika, S., Akhil, K.H., Deepa, K.: ML assisted sign language to speech conversion gloves for the differently abled. In: 2024 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI), Gwalior, India, pp. 1–6 (2024). https://doi.org/10.1109/IATMSI60426.2024.10503002
5. Shashidhar, R., Hegde, S.R., Chinmaya, K., Priyesh, A., Manjunath, A.S., Arunakumari, B.N.: Indian sign language to speech conversion using convolutional neural network. In: 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon), Mysuru, India, pp. 1–5 (2022). https://doi.org/10.1109/MysuruCon55714.2022.9972574
6. Kowsigan, M., Dhawan, R., Kundu, A.: An efficient speech to sign language conversion and text recognition through live gesture. In: 2024 IEEE International Conference on Smart Power Control and Renewable Energy (ICSPCRE), Rourkela, India, pp. 1–6 (2024). https://doi.org/10.1109/ICSPCRE62303.2024.10674922
7. Peguda, J., Santosh, V.S.S., Vijayalata, Y., Deepa, A., Mounish, V.: Speech to sign language translation for Indian languages. In: 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, pp. 1131–1135 (2022). https://doi.org/10.1109/ICACCS54159.2022.9784996
8. Om Kumar, C.U., Devan, K.P.K., Renukadevi, P., Balaji, V., Srinivas, A., Krithiga, R.: Real time detection and conversion of gestures to text and speech to sign system. In: 2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, pp. 73–78 (2022). https://doi.org/10.1109/ICESC54411.2022.9885562
9. Rai, D., Rana, N., Kotak, N., Sharma, M.: Real-time speech to sign language translation using machine and deep learning. In: 2024 11th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, pp. 1–5 (2024). https://doi.org/10.1109/ICRITO61523.2024.10522437

10. Nurfita Sari, V.E., Arifin, A., Arrofiqi, F.: Design of two-way indonesian sign language system based on smart-glove with artificial neural network classification method. In: 2024 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM), Surabaya, Indonesia, pp. 1–6 (2024). https://doi.org/10.1109/CENIM64038.2024.10882818
11. Gunvantray, T.D., Ananthan, T.: Sign language to text translation using convolutional neural network. In: 2024 International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, pp. 1–5 (2024). https://doi.org/10.1109/ESCI59607.2024.10497209
12. Thong, S.X., Tan, E.L., Goh, C.P.: Sign language to text translation with computer vision: bridging the communication gap. In: 2024 3rd International Conference on Digital Transformation and Applications (ICDXA), Kuala Lumpur, Malaysia, pp. 215–219 (2024). https://doi.org/10.1109/ICDXA61007.2024.10470532
13. Mannepula, S., Hrishitha, M., Reddy, R.R., China Ramu, S.: Sign language-to-emotion-specific text translation: a review. In: 2024 3rd International Conference on Automation, Computing and Renewable Systems (ICACRS), Pudukkottai, India, pp. 738–746 (2024). https://doi.org/10.1109/ICACRS62842.2024.10841673
14. Seviappan, A., Ganesan, K., Anbumozhi, A., Reddy, A.S., Krishna, B.V., Reddy, D.S.: Sign language to text conversion using RNN-LSTM. In: 2023 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI), Chennai, India, pp. 1–6 (2023). https://doi.org/10.1109/ICDSAAI59313.2023.10452617

Automated Papaya Fruit Classification Using CNN Models

Rupa Lalam, Premkumar Borugadda, K. Lavanya, and Vinoda Nadella

Abstract Every year, a large number of papaya farmers suffer significant losses due to diseases affecting their crops. Unfortunately, many of these farmers lack the knowledge and tools to detect these issues early on. Often, by the time the disease is noticed, the fruits are already damaged and can't be saved. Because of this recurring problem, some farmers have even become hesitant to grow papaya again. To help address this issue, researchers have turned to deep learning technologies to develop a system that can automatically detect and classify the condition of papaya fruits. In this study, papaya samples were collected from farms in Vijayawada, Andhra Pradesh. The goal was to classify the fruits into three categories: unripe, ripe, and defective. A dataset containing 4500 images (1500 each category) was created. These images were pre-processed through steps like resizing, normalization, and label encoding to get them ready for model training. Several deep learning models were developed from scratch, including CNN, AlexNet and VGG16, to perform the classification task. Among these, a custom CNN model with 12 convolutional layers and 3 fully connected layers delivered the best results, achieving a test accuracy of 94.22%. This research demonstrates how deep learning can play a key role in automating agricultural processes, especially for tasks like fruit classification and quality checking.

R. Lalam (✉) · K. Lavanya
Department of Processing and Food Engineering, Dr. NTR College of Agricultural Engineering, Bapatla, Andhra Pradesh, India
e-mail: rupalalam2001@gmail.com

K. Lavanya
e-mail: k.lavanya@angrau.ac.in

P. Borugadda
Deparment of Computer Science and Engineering, SRM University-AP, Amaravati, Andhra Pradesh, India
e-mail: premkumar.b@srmap.edu.in

V. Nadella
Department of Food Process Engineering, Dr. NTR College of Food Science and Technology, Bapatla, Andhra Pradesh, India
e-mail: n.vinoda@angrau.ac.in

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_7

Keywords Deep learning · Convolutional layer · Fully connected layer · Pre-processing · Scratch

1 Introduction

Papaya (*Carica papaya*) is a tropical fruit and is highly nutritious, rich in vitamins C and A. In 2022–23 the papaya production in India is 6.56 million metric tons and in Andhra Pradesh it was 978,000 metric tons. The papaya tree typically grows between 2 and 10 m tall. Its fruit is well known for its naturally sweet flavor, vibrant color, and versatility, making it easy to include in various dishes. Defective fruits cause economic loss and may affect health. Early detection of defects in fruits is crucial for preserving their quality, maintaining nutritional value, ensuring consumer satisfaction and preventing financial losses for producers.

In recent times, image processing techniques have been employed in various stages like cultivation, harvesting, and post-harvest management to enhance productivity, quality assessment, and disease detection. Determining the maturity status of fruits is crucial for assessing their eating quality and deciding the appropriate storage duration before consumption [1]. Identifying the maturity of papaya fruit will greatly support farmers to avoid under-matured or over-matured papaya harvesting. Computer vision generates detailed descriptions of physical objects based on images [2], while image processing can reveal finer details like shape and color. Because color plays a critical role in fruit quality, defects may go unnoticed during manual sorting. Automated systems, on the other hand, can detect these defects, improving quality and boosting economic value. Removal of noise is done in the pre-processing to attain high quality features [3]. This research offers a practical solution for the agriculture sector by introducing an automated deep learning system that helps farmers accurately identify the quality of papaya fruits, reducing losses and supporting more confident crop management decisions.

The deep learning techniques, convolution neural networks are a commonly employed method in image-based data applications. CNNs are a type of deep learning algorithms primarily used for image recognition and processing tasks. They are particularly effective for tasks involving visual data because they can learn spatial hierarchies, or patterns within images, in a way that makes them highly suitable for recognizing objects, faces, and even minute details.

Here the present study undertaken to classify the defects and maturity categories based on the extracted images features of the papaya fruits using the customized CNN model, Alexnet, VGG16.

This work is justified by the growing need for efficient, technology-driven solutions in agriculture. By using image processing and deep learning, farmers can now identify papaya maturity and defects more accurately and at the right time. This not only boosts productivity but also helps minimize losses and improve fruit quality for the market.

1.1 Objectives

1. To create an image database for different maturity levels and defects in quality of papaya fruits
2. To develop a custom CNN, AlexNet, VGG16
3. To compare the efficiency of developed algorithms.

1.2 Limitations

- Needs a Lot of Data to Learn Well: When you build a model from the ground up, it usually needs no. of examples to really understand patterns. If the dataset is small, the model may just memorize what it sees instead of actually learning.
- No Augmentation, No Robustness: Without adding variations like flips, rotations, or noise to your training data, the model only learns one version of reality. As a result, it might struggle when faced with even small changes in new data.
- Tends to Overfit on Small Datasets: Training without techniques like dropout or early stopping, the model can easily become too confident about the training data and completely miss the mark on test or real-world inputs.
- Takes More Time and Power to Train: Starting from scratch means longer training times and heavier use of hardware like GPU. It can be slow and expensive compared to using existing models or transfer learning techniques.

2 Literature Review

Chen et al. [4] introduced a novel image dataset with 23,158 examples across nine classes of papaya fruit diseases, along with a robust disease detector named Yolo-Papaya. This detector is based on the YoloV7 detector and incorporates a convolutional block attention module (CBAM) attention mechanism. Yolo-Papaya achieved an overall mean average precision (mAP) of 86.2%, with performance exceeding 98% in categories like "healthy fruits" and "Phytophthora blight." This detector and dataset are suitable for practical fruit quality control applications and provide a strong benchmark for papaya fruit disease detection. The dataset and source code are publicly available on the project page, promoting study reproducibility and research advancement in this field.

De Moraes et al. [5] developed an efficient conveyor system supported by image processing and a transfer learning approach. A dataset comprising 1109 images of over-mature papayas, 1054 images of mature papayas, and 1367 images of immature papayas was used to train a CNN. Models including EfficientNetV2B1, MobileNetV3, ResNetRS50, and VGG19 were employed for training. Among these, MobileNetV3 demonstrated the best performance, achieving 100% accuracy within 10 epochs and a loss of 0.006%. EfficientNetV2B1 also reached 100% accuracy but

had a higher loss of 0.31%. ResNetRS50 similarly achieved 100% accuracy with a loss of 0.54%. VGG19 showed the lowest performance, achieving 100% accuracy only after 30 epochs, with a loss of 151.37%. Therefore, MobileNetV3 was identified as the most accurate model for classifying the maturity status of papaya fruits.

Behera et al. [6] proposed a model for classification of papaya fruits based on maturity by a deep learning model. The dataset included 300 high-resolution images categorized into three classes: Mature (Class 0), Partially Mature (Class 1), and Unmature (Class 2). To improve computational efficiency without affecting quality, the images were resized to 128 $\times$ 128 pixels and labeled manually. The model used VGG16, a convolutional neural network (CNN) known for object recognition. Compared to earlier models like AlexNet, VGG16 improves accuracy by using multiple 3 $\times$ 3 convolutional filters instead of larger ones. This method enhances feature extraction and pattern recognition. VGG16 eliminated the need for manual feature extraction and achieved 100% accuracy with a training time of 1 min and 52 s. While deep learning models typically require large datasets and longer training times, this model effectively classified papaya maturity with high precision.

Agarwal et al. [7] stated that deep learning algorithms for predicting fruit maturity and quality, specifically focusing on the shelf life of bananas. Two datasets were used: a custom dataset of 2100 banana images categorized as ripe, unripe, and over-ripe (with 700 images per category) and the Fruit 360 dataset from Kaggle. To expand the dataset, image augmentation techniques were applied, increasing the dataset size to 18,900 images. Models used both a custom-built convolutional neural network and the AlexNet architecture to analyse multiple datasets. Custom dataset, the CNN performed exceptionally well, reaching an accuracy of 98.25%, while AlexNet achieved 81.75%. An augmented version of the same dataset, the results improved even further—CNN reached 99.36% accuracy, and AlexNet slightly edged it out with 99.44%. For comparison, tested both models on the Fruit 360 dataset, where CNN achieved 81.96% and AlexNet scored 81.75%. Overall, these results highlight that our custom CNN model consistently delivered the best performance in classifying banana ripeness and assessing fruit quality across all three datasets.

Mundhada et al. [8] explored how artificial intelligence, especially deep learning, can support the agricultural sector—particularly in the handling of fruits and vegetables. With agriculture playing a vital role in economic development, the increasing demand for fresh, ripe produce has made it necessary to adopt advanced technologies. Deep learning, and specifically Convolutional Neural Networks (CNNs), have shown great potential in analysing images and identifying key features. Their study focused on classifying fruits by ripeness using a dataset of 9997 images from 15 different fruit types. By applying deep learning techniques, they achieved an accuracy of 90.24% in fruit detection and maturity grading, highlighting the effectiveness of AI in improving quality control and efficiency in fruit production.

Risdin et al. [9] introduced an efficient method for detecting fruits using deep convolutional neural networks (CNNs). Their goal was to create a fast, accurate, and reliable system that could identify various fruit types using machine learning. Since recognizing fruit images can be challenging due to their diversity, CNNs were used to automatically extract features and improve detection accuracy. A dataset

Table 1 A list of advancements involving papaya through the use of deep learning

Author and year	Model	Fruit	Accuracy (%)
De Moraes et al. [5]	YoloV7	Papaya fruits (healthy fruit and Phytophthora blight)	86.2
Jayabandu et al. [10]	MobileNetV3	Papaya fruits (over mature, immature, mature)	100
Masawabe et al. [2]	VGG16	Papaya fruits (partially mature, unmature, mature)	100
Aherwadi et al. [11]	AlexNet	Bananas (maturity and quality)	99.44
Mundhada et al. [8]	CNN	Fruits (defect and 3 stages of maturity)	90.24
Risdin et al. [9]	CNN	Fruits (four different classes)	99.89

made from commonly found fruits was used to test the model, and CNNs clearly outperformed traditional approaches like support vector machines that rely on manual feature extraction. The model was later retrained using 2403 images of four different fruit categories, all captured with a smartphone camera. It achieved an impressive accuracy of 99.89%, showing its strong potential for real-world fruit recognition tasks (Table 1).

3 Methadology

This section discusses about the papaya classification framework, datasets and hardware configuration details.

Figure 1 shows the classification process, which is carried out in five main steps: collecting the data, preparing it through preprocessing, extracting important features, classifying the data, and finally, evaluating how well the trained models perform.

3.1 Dataset

Data plays a pivotal role in any research. In this study, the benchmark dataset utilized is the papaya dataset, which is sourced from the papaya farm collection and has a total size of approximately 11.4 GB. The papaya dataset comprises 4500 images representing 3 different papayas, like unripen, ripen, defect as shown in Fig. 2.

This research specifically focuses on images of papaya fruits. The dataset provides a different set of images classify into three classes of papaya fruits. Table 2 illustrates the distribution of images across each class the unripen images of 1500 and ripen category of 1500 and finally the defect papaya images of 1500 and for training,

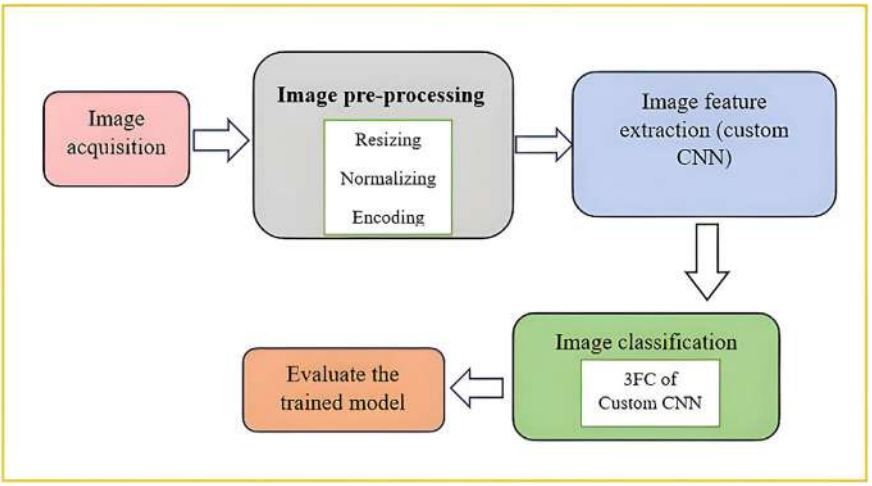


Fig. 1 Framework architecture

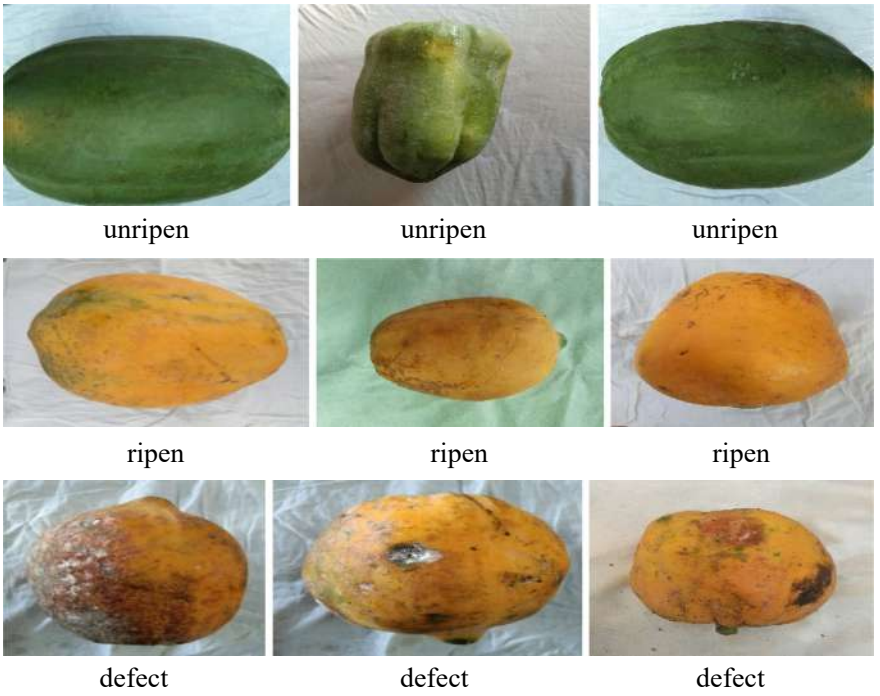


Fig. 2 Different classes of papaya images

Table 2 Dataset

S. No.	No. of classes	No. of images
1	Unripen	1500
2	Ripen	1500
3	Defect	1500
	Total images	4500

Table 3 Dataset structure used in models

S. No.	Name of the classes	Number of training images	Number of validation images	Number of testing images	Total number of images
1	Defected	1200	150	150	1500
2	Ripen	1200	150	150	1500
3	Unripen	1200	150	150	1500
Total images		3600	450	450	4500

validation and testing of model the dataset distribution of papayas was shown in Table 3.

3.2 Image Pre-processing

The initial phase involves preprocessing the input image data, which has a size of (height, width, color) = 227 * 227 * 3. This is accomplished through a series of operations. The dataset classes are first labelled using label encoding, followed by the application of a one-hot encoding technique. The class labels include unripen papaya, ripen papaya, defected papaya which are originally in text form. Since text data cannot be directly interpreted by machine learning models, it needs to be transformed into numerical format. Label encoding is utilized to assign integer values ranging from 0 to n − 1, where n represents the total number of classes (n = 3), based on their alphabetical order. one-hot encoding is employed as a method to handle categorical variables, creating new attributes corresponding to the unique values of the categorical data [12]. Finally, the pixel values of the images are scaled to a range between 0 and 1 to normalize the data for better model performance.

3.3 Feature Extraction

In the feature extraction stage, important details from papaya images are converted into numerical values that can be used for classification. The architecture of the

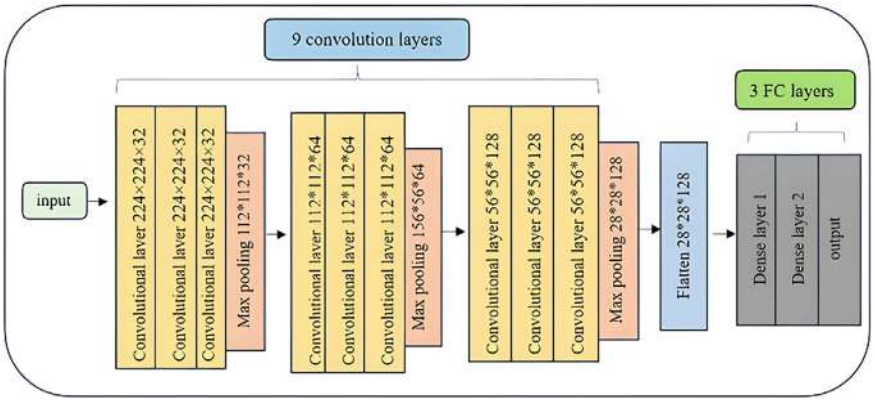


Fig. 3 Architecture of custom CNN model

custom CNN model, illustrated in Fig. 3, includes 9 convolutional layers dedicated to extracting features from images of unripe, ripe, and defective papayas during the training phase. For classification, the model uses 3 fully connected (FC) layers. Each input image is resized to 224 × 224 pixels and passed through convolutional layers using 3 × 3 filters. These filters help in identifying patterns and textures within the images. To maintain the image dimensions during convolution, same padding is applied, and the ReLU activation function is used after each convolution to introduce non-linearity. Max pooling layers follow some of these convolution steps to down sample the image, reducing its size while keeping the essential features. At the final stage, a softmax activation function is used in the output layer to classify the images based on the probability of each category. This structure allows the deep learning model to effectively process and categorize the papaya fruits after extracting relevant features.

The Alexnet had the five convolution layers to extracts the features for better model learning, and 3 fully connected layers for classification of the given 3 classes of papaya fruits (defect, ripen, unripen) with hep of softmax activation function to obtain in probability for of classification as shown in Fig. 4. Similarly, for the VGG16 has the 13 convolutional layers and 3 fully connected layers for extraction of features and classification of different classes with great accuracy score as shown in Fig. 5.

3.4 Classification of Images

In image classification, the fully connected layers in each model play an important role in making the final decision. These layers use the features extracted by earlier layers and, with the help of activation functions, different activation function is used like softmax mostly used for finding in probability ratio to classify the image into the correct category. In this work, the custom CNN is designed with 3 FC layers to

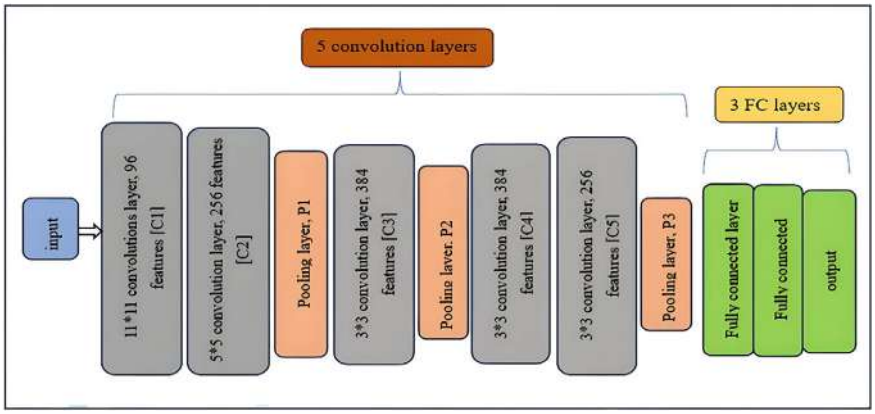


Fig. 4 Architecture of AlexNet model

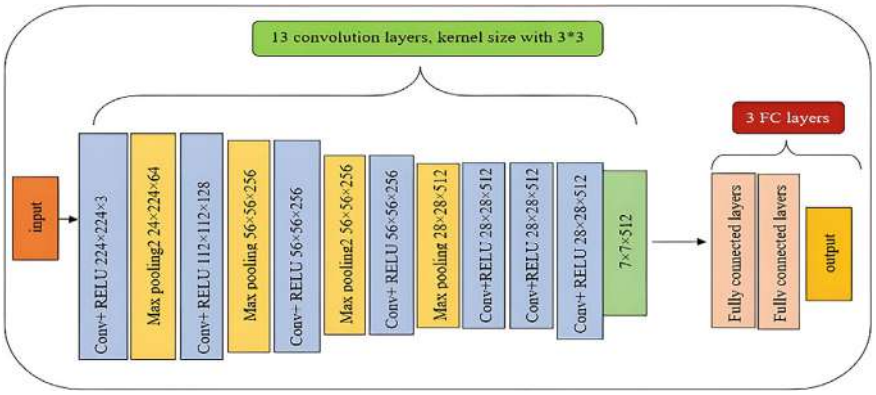


Fig. 5 Architecture of scratch VGG16 model

classify papayas into three different classes. Similarly, both AlexNet and VGG16 also include 3 FC layers at the end of their architectures. Among these models, the one that gives the highest accuracy is considered the best for classifying the papaya images.

3.5 Evaluation Metrics

For both binary and multi-class classification tasks, a confusion matrix is critical for analysing model performance. In multi-class classification as shown in Table 4, where the dataset contains more than two classes, the evaluation metrics for each

Table 4 Confusion matrix for multi class classification

	Predicted classes				
Actual classes		C1	C2	…	CK
	C1	**TP1**			
	C2		**TP2**		
	…	…	…	**…**	…
	CK				**TPK**

The "TP" means true positive

class must be calculated individually. The overall performance is then aggregated using specific averaging techniques.

To evaluate the effectiveness of classification models, metrics like "weighted-average precision, recall, and F1-Score" are frequently employed [13, 14]. Performance measures, including Micro-averaging, Macro-averaging, and Weighted-averaging, are widely used.

Common Averaging Method

1. **Macro-averaging (MA)**: It works by computing the metric separately for each class and then averaging the results without giving any class extra weight. So, every class is treated equally, no matter how many samples it has.

$$\text{MAPrecision} = \frac{\text{TP}_\text{A}/(\text{TP}_\text{A} + \text{FP}_\text{A}) + \text{TP}_\text{B}/(\text{TP}_\text{B} + \text{FP}_\text{B}) + \text{TP}_\text{C}/(\text{TP}_\text{C} + \text{FP}_\text{C})}{\text{N}} \tag{1}$$

$$\text{MARecall} = \frac{\text{TP}_\text{A}/(\text{TP}_\text{A} + \text{FN}_\text{A}) + \text{TP}_\text{B}/(\text{TP}_\text{B} + \text{FN}_\text{B}) + \text{TP}_\text{C}/(\text{TP}_\text{C} + \text{FN}_\text{C})}{\text{N}} \tag{2}$$

$$\text{MAF1\_Score} = \frac{\text{F1 score}_\text{A} + \text{F1 score}_\text{B} + \cdots \text{F1 score}_\text{K}}{\text{N}} \tag{3}$$

N = Number of classes.

4 Results

4.1 Experimental Setup

See Table 5.

Table 5 Hardware configurations

S. No.	Hardware and software	Features
1	Memory (RAM)	8.00 GB
2	Processor	11th Gen Intel(R) Core (TM) i5-1155G7 @ 2.50 GHz 2.50 GHz
3	Operating system	Windows 11 and 64 bits
4	Integrated development environment (IDE)	Googlecolab

Table 6 Hyperparameters of CNN models

S. No.	Hyperparameters	Optimal values at FC of CNN	Optimal values at FC of VGG16	Optimal values at FC of AlexNet
1	No. of dense layers	3	3	3
2	No. of neurons in dense layers	1024, 1024	4096, 4096	4096, 4096
3	Activation function at dense layer	Relu	Relu	Relu
4	Activation function at output layer	Soft max	Soft max	Soft max
5	Optimizers	SGD	SGD	SGD
6	Learning rate	0.0001	0.001	0.001
7	Dropout	0.5	0.5	0.5

4.2 Hyperparameters Tuning

See Table 6.

4.3 Results

4.3.1 Custom CNN

This section presents the analysis of results obtained from a custom CNN model comprising 9 convolutional layers and 3FC layers. The model was specifically developed to classify papaya images using a balanced dataset. The classification process was carried out by the FC layers, which transformed extracted features. No dimensionality reduction techniques were applied during the evaluation.

The CNN architecture consists of 9 convolutional layers responsible for feature extraction, followed by 3 FC layers that facilitated classification. The model training and validation performance are illustrated in Fig. 6a, b, showing the loss and accuracy of the custom CNN model. During training, the model achieved an accuracy of

95.92%, with a corresponding loss of 11.26%. The validation phase resulted in an accuracy of 93.33% and a loss of 14.34%. The test accuracy of the model was determined to be 94.22%.

Confusion matrix for the CNN model with 3 fully connected layers is illustrated in Fig. 7. Out of 450 validation images, the model correctly classified 424 samples, while 26 samples were misclassified. The classification report shown in Fig. 13a for the CNN model presents a comprehensive evaluation of its performance in papaya image classification. It covers important measures like precision, recall, and the F1 score, which help us understand how accurate the model is and how well it performs overall. These metrics help assess how well the model identifies and classifies papaya images, ensuring a detailed understanding of its strengths and areas for improvement.

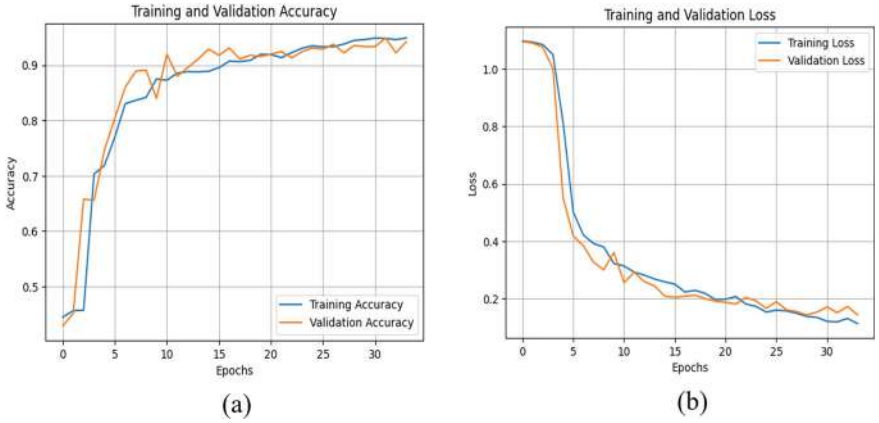


Fig. 6 **a** Training and validation accuracy, **b** training and validation loss

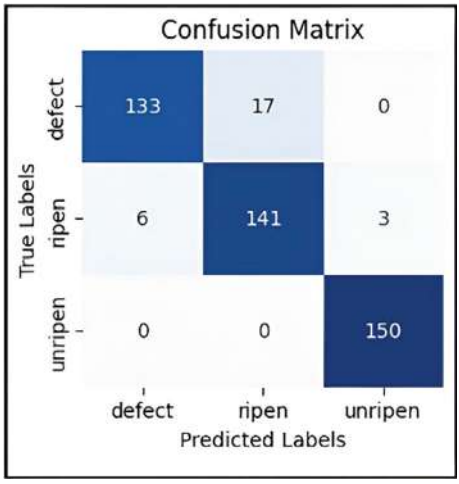


Fig. 7 Confusion matrix on CNN from scratch

4.3.2 AlexNet

The experimental results were conducted, where the AlexNet model was built from the ground up. It was trained using a well-balanced benchmark papaya dataset to effectively extract essential features for classification. The classification process utilized the fully connected layers of AlexNet for feature extraction. The assessment was conducted on the performance of the classification model without applying dimensionality reduction.

AlexNet comprises 5 convolutional layers, and its 3 fully connected layers were utilized to extract meaningful features from papaya images. Figure 8a, b describes the training and validation loss curve, and the training and validation accuracy of the AlexNet model. The model achieved a training accuracy of 75.11%, with a corresponding training loss of 45.45%. Furthermore, the AlexNet model achieved a validation accuracy of 75.56% and a validation loss of 42.01%. The overall test accuracy of the model was recorded at 75.55%, indicating its effectiveness in classifying papaya images.

Confusion matrix for the 3FC of AlexNet is presented in Fig. 9. Among 450 validation image samples, 340 were correctly predicted (CP), while 110 were incorrectly predicted (WP). The classification report for the 2FC of AlexNet without dimensionality reduction is provided in Fig. 13b. This table outlines the accuracy, precision, recall, and F1 score of the developed AlexNet model with macro average of each class precision, recall, F1score of papaya fruit. These classification report help to assess how well the model identifies and classifies papaya images, ensuring a detailed understanding areas for improvement.

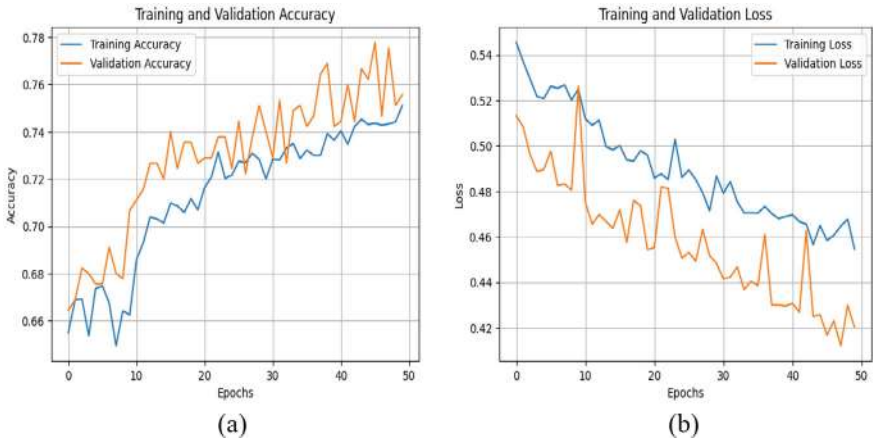


Fig. 8 **a** Accuracy of training and validation, **b** loss of training and validation dataset

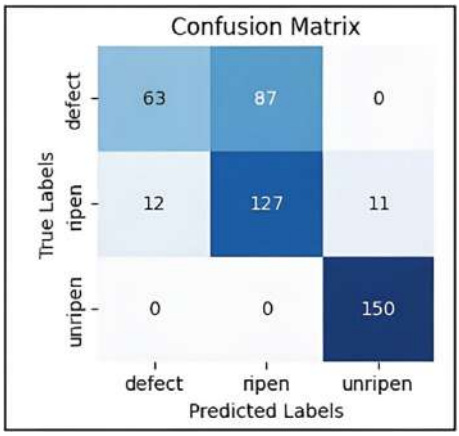


Fig. 9 Confusion matrix on AlexNet from scratch

4.3.3 VGG16

VGG16 has 13 convolutional neurons and in that 3FC are utilized for the features extraction of the papaya images for useful information. Figure 10a, b illustrates the training and validation loss curves, along with the training and validation accuracy of the 3FC VGG16 classification model. The model achieved a highest training accuracy of 97.92%, with a corresponding training loss of 4.93%. Additionally, the validation accuracy and validation loss were recorded as 89.33% and 36.42%, respectively. The final test accuracy of the VGG16 model is 90.88%.

The confusion matrix for the 3FC of VGG16 is presented in Fig. 11. Among 450 validation image samples, 409 were correctly predicted (CP), while 41 were incorrectly predicted (WP). The classification report for the 3FC of VGG16 without dimensionality reduction is provided in Fig. 13c. This table outlines the accuracy,

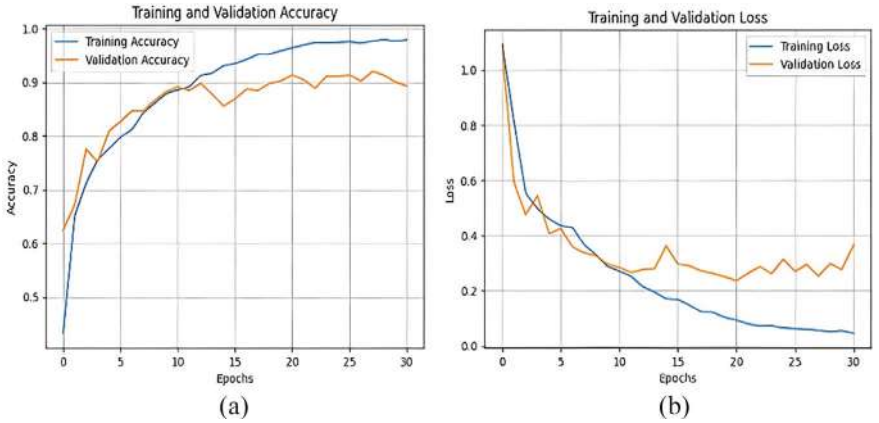


Fig. 10 **a** Accuracy of training and validation, **b** loss of training and validation

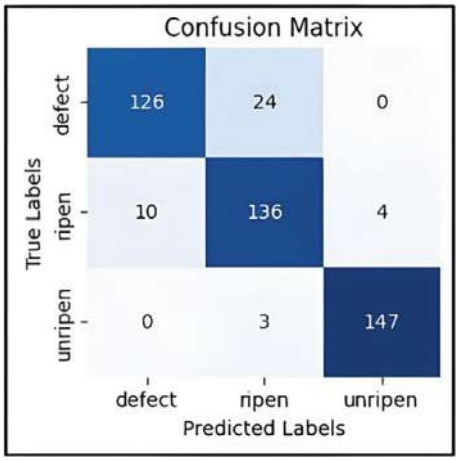


Fig. 11 Confusion matrix on VGG16 from scratch

precision, recall and F1 score of the developed VGG16 model with weighted average of each class precision, recall, F1score of papaya fruit.

4.4 Evaluating and Comparing the Accuracy of CNN Models

4.4.1 Comparison of Accuracy of the CNN Models

The deep learning models developed from scratch were evaluated for classifying three different categories of papaya fruits. The models implemented, including CNN, AlexNet and VGG16, were assessed based on important performance measures like accuracy, precision, recall and the F1 score are calculated using the values from the confusion matrix. These metrics help assess how well the model is making predictions. A comparative analysis of these metrics was conducted to identify the most effective model for papaya fruit classification.

When a model shows very high accuracy during training (95%) but performs much worse on validation (70%) and testing data (68%), it's a sign of overfitting. This means the model has learned the training data too well—almost like memorizing it—and can't handle new, unseen data effectively. Conversely, the training accuracy is 90%, with validation and testing accuracy close at 88% and 87%, respectively, the model demonstrates good generalization, performing consistently across all datasets.

Table 7 presents a detailed comparison of the Custom CNN, AlexNet, and VGG16 models, highlighting their training and testing accuracies, along with the total number of parameters-including both trainable and non-trainable ones. The table also reflects how each model performed in classifying papayas into the three categories, showing both correct and incorrect predictions out of a test set of 450 images. This analysis

helps in understanding how well each model generalizes and handles real-world classification of papaya fruits.

In Fig. 12 bar chart highlights the performance comparison between three deep learning models Custom CNN, AlexNet, and VGG16 used to classify papaya fruits into three categories. Among them, the Custom CNN model outperformed the others, achieving the highest test accuracy of 94.22%, followed closely by VGG16 and then AlexNet. Figure 13a–c presents the detailed classification reports for each model, including metrics such as accuracy, precision, recall, and F1-score, showcasing their effectiveness in supporting better papaya cultivation through accurate fruit classification.

Table 7 Performance comparison of CNN models

Model	Tr.A (%)	Te.A (%)	V.S (450)		T.P	Tr.P	N.T.P
			CP	WP			
CNN	94.22	**94.22**	424	26	104,294,915	104,294,915	0
AlexNet	75.11	75.56	340	110	46,760,707	46,760,003	704
VGG16	97.92	90.88	409	41	268,545,671	134,272,835	0

Note Tr.A training accuracy, *Te.A* test accuracy, *V.S* validation samples, *C.P* correct predictions, *W.P* wrongly predicted, *T.P* total parameters, *Tr.P* trainable parameters, *N.T.P* non-trainable parameters

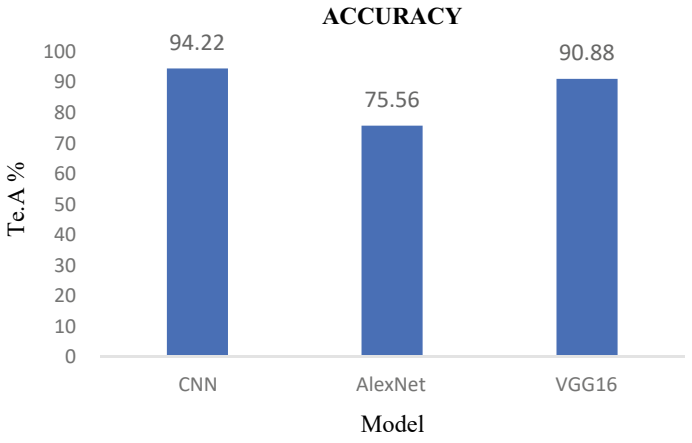


Fig. 12 Performance comparison of CNN model

Classes	Precision	Recall	F1_Score	Support
Defected papaya	0.9568	0.8867	0.9204	150
Ripen papaya	0.8924	0.9400	0.9156	150
Unripen papaya	0.9804	1.0000	0.9901	150
Accuracy			0.9422	450
Macro AVG	0.9432	0.9422	0.9420	450
Weighted AVG	0.9432	0.9422	0.9420	450

(a)

Classes	Precision	Recall	F1_Score	Support
Defected papaya	0.8400	0.4200	0.5600	150
Ripen papaya	0.5935	0.8467	0.6978	150
Unripen papaya	0.9317	1.0000	0.9646	150
Accuracy			0.7556	450
Macro AVG	0.7884	0.7556	0.7408	450
Weighted AVG	0.7884	0.7556	0.7408	450

(b)

Classes	Precision	Recall	F1_Score	Support
Defected papaya	0.9265	0.8400	0.8811	150
Ripen papaya	0.8344	0.9067	0.8690	150
Unripen papaya	0.9735	0.9800	0.9767	150
Accuracy			0.9090	450
Macro AVG	0.9114	0.9089	0.9090	450
Weighted AVG	0.9114	0.9089	0.9090	450

(c)

Fig. 13 Classification reports of **a** custom CNN, **b** AlexNet, **c** VGG16

5 Conclusion

The results clearly show that the custom CNN model performed the best among the three, achieving a test accuracy of 94.22%. This indicates that the model is highly effective in accurately identifying the ripeness and defects in papaya fruits. While VGG16 also showed strong performance with an accuracy of 90.88%, it still fell slightly short of the custom CNN. On the other hand, AlexNet lagged behind with an accuracy of 75.56%, suggesting it may not be as suitable for this specific classification task. Overall, this comparison highlights the importance of designing task-specific deep learning models for better accuracy and performance in agricultural applications. An automated deep learning-based tool that assists farmers in accurately identify papaya fruit quality. By enabling early detection and classification, the system can help reduce crop losses and encourage farmers to cultivate papaya with greater confidence.

6 Future Research Direction

This research can be further strengthened expand the dataset by collecting more diverse images of papaya fruits. This would help the model generalize better across real-world scenarios. To address the current limitations in variety, applying data augmentation techniques such as image rotation, flipping, and brightness adjustment can artificially increase the size and diversity of the dataset without additional manual effort. Principal Component Analysis (PCA). Additionally, building a lightweight, real-time detection system in the form of a mobile app or smart device would greatly benefit farmers, giving them instant feedback in the field. Future models could also be trained to detect not just ripeness and defects, but also common papaya diseases, offering a more complete health diagnosis. Combining different deep learning models or hybrid approaches with traditional machine learning methods may further improve performance.

Acknowledgements I would like to express my heartfelt gratitude to Dr. B. Premkumar for his valuable guidance, support, and motivation throughout the course of this work. His encouragement and insights have been instrumental in shaping this effort. I also wish to sincerely thank Dr. K. Lavanya and Dr. N. Vinoda for their continued support, suggestions, and encouragement, which greatly contributed to the progress of this work. My sincere appreciation goes to Dr. NTR College of Agricultural Engineering, Department of Processing and Food Engineering, for providing the necessary resources that made this work possible.

References

1. Magwaza, L.S., Opara, U.L.: Analytical methods for determination of sugars and sweetness of horticultural products—a review. Sci. Hortic. **184**, 179–192 (2015)
2. Al-Masawabe, M.M., Samhan, L.F., AlFarra, A.H., Aslem, Y.E., Abu-Naser, S.S.: Papaya Maturity Classifications Using Deep Convolutional Neural Networks (2021)
3. Lalam, R., Lavanya, K., Nadella, V., Kiran, B.R.: Automatic sorting and grading of fruits based on maturity and size using machine vision and artificial intelligence. J. Sci. Res. Rep. **31**(1), 153–163 (2025)
4. Chen, S., Xiong, J., Jiao, J., Xie, Z., Huo, Z., Hu, W.: Citrus fruits maturity detection in natural environments based on convolutional neural networks and visual saliency map. Precis. Agric. **23**(5), 1515–1531 (2022)
5. De Moraes, J.L., de Oliveira Neto, J., Badue, C., Oliveira-Santos, T., de Souza, A.F.: Yolo-papaya: a papaya fruit disease detector and classifier using CNNs and convolutional block attention modules. Electronics **12**(10), 2202 (2023)
6. Behera, S.K., Rath, A.K., Sethy, P.K.: Maturity status classification of papaya fruits based on machine learning and transfer learning approach. Inf. Process. Agric. **8**(2), 244–250 (2021)
7. Agarwal, N., Sondhi, A., Chopra, K. and Singh, G.: Transfer learning: survey and classification. In: Smart Innovations in Communication and Computational Sciences: Proceedings of ICSICCS 2020, pp. 145–155 (2021)
8. Mundhada, H., Sanagavarapu, S., Sood, S., Damdoo, R., Kalyani, K. (2023). Fruit detection and three-stage maturity grading using CNN. Int. J. Next Gen. Comput. **14**(1).
9. Risdin, F., Mondal, P.K., Hassan, K.M.: Convolutional neural networks (CNN) for detecting fruit information using machine learning techniques. IOSR J. Comput. Eng. **22**(2), 1–13 (2020)

10. Jayabandu, L.P.A., Samarasinghe, P.P.M., Wanniarachchi, W.N.N., Atapattu, H.Y.R.: An automated system to classify the maturity status of papaya fruits based on transfer learning approach (2023)
11. Aherwadi, N., Mittal, U., Singla, J., Jhanjhi, N.Z., Yassine, A., Hossain, M.S.: Prediction of fruit maturity, quality, and its life using deep learning algorithms. Electronics **11**(24), 4100 (2022)
12. Al-Shehari, T., Alsowail, R.A.: An insider data leakage detection using one-hot encoding, synthetic minority oversampling and machine learning techniques. Entropy **23**(10), 1258 (2021)
13. Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. Inf. Process. Manage. **45**(4), 427–437 (2009)
14. Borugadda, P., Lakshmi, R., Sahoo, S.: Transfer learning VGG16 model for classification of tomato plant leaf diseases: a novel approach for multi-level dimensional reduction. Pertanika J. Sci. Technol. **31**(2) (2023)

Seamless EV Charging Through GPS-Guided Vehicle-to-Vehicle Power Transfer and Wireless Charging Lanes

S. P. Vimal, S. Sivanika Sri, S. Trisha, and M. Vijaya Manogna

Abstract This project revolutionizes electric vehicle (EV) charging by integrating GPS-enabled vehicle-to-vehicle (V2V) wireless power transfer with strategically deployed wireless charging lanes, offering a significant departure from conventional static methods. Through a dedicated user portal, EV owners can request on-demand charging assistance, activating precise GPS tracking of their location and facilitating the dispatch of mobile charging units or guidance to wireless charging infrastructure. Notably, the inclusion of wireless charging lanes allows EVs to automatically initiate charging upon nearing designated spots, seamlessly extending their range and reducing dependence on stationary infrastructure. This dual approach of mobile and in-road wireless charging significantly enhances the accessibility and convenience of EV charging, promising to overcome range anxiety and accelerate the adoption of electric vehicles.

Keywords Vehicle to vehicle communication · GPS technology · Wireless power charging

S. P. Vimal (✉) · S. Sivanika Sri · S. Trisha · M. Vijaya Manogna
Department of ECE, Sri Ramakrishna Engineering College, Coimbatore, Tamil Nadu, India
e-mail: vimal.sp@srec.ac.in

S. Sivanika Sri
e-mail: sivanikasri.2102214@srec.ac.in

S. Trisha
e-mail: trisha.2102236@srec.ac.in

M. Vijaya Manogna
e-mail: vijayamanogna.2102248@srec.ac.in

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*,
Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_8

1 Introduction

Addressing the growing adoption of electric vehicles (EVs) and the ongoing issues of range anxiety and charging accessibility in Coimbatore's vibrant urban and peri-urban environment, this project presents a distinctive and thorough solution: a synergistic dual-system of GPS-enabled on-demand Vehicle-to-Vehicle (V2V) wireless power transfer, strategically combined with a forward-looking strategy for future wireless charging lane integration. This innovative method transcends the drawbacks of traditional static charging infrastructure by providing an immediate, flexible, and location-sensitive charging model tailored specifically to the unique requirements of Coimbatore's developing transportation ecosystem. The foundation of this initiative is the creation of an intuitive mobile application utilizing accurate GPS technology. This enables EV owners experiencing energy depletion to effortlessly request a charge, relaying their precise location within Coimbatore to a centralized service platform. This activates the intelligent dispatch of strategically placed mobile charging provider EVs, successfully turning charging into a dynamic, on-demand service delivered straight to the user. Equipped with robust wireless power transmission systems, these mobile units offer a convenient and adaptable energy transfer solution across various real-world situations, from navigating busy urban thoroughfares to reaching vehicles in more remote peri-urban areas surrounding Coimbatore. This proactive V2V charging capability signifies a notable improvement over fixed charging stations, delivering unmatched user convenience and directly alleviating range anxiety within the local context. Furthermore, acknowledging the long-term direction of sustainable transportation, this project strategically includes a research-based framework for the future incorporation of wireless charging lanes into Coimbatore's infrastructure. While the immediate focus centers on the deployable V2V system, our detailed analysis of existing and emerging wireless charging technologies provides a thoughtful pathway for potential infrastructure development. This future-oriented perspective guarantees that this initiative not only meets current charging requirements but also strategically aligns with and anticipates the advancement of seamless electric mobility solutions within the region. The distinctiveness and professional rigor of this project reside in its holistic, dual-pronged strategy, meticulously designed to the geographic and infrastructural specifics of Coimbatore, offering an immediate and flexible V2V charging solution while simultaneously establishing a well-informed vision and conceptual foundation for future seamless integration with dynamic wireless charging infrastructure. This comprehensive and adaptable approach positions Coimbatore as a potential leader in the implementation of a truly seamless and sustainable electric mobility ecosystem.

2 Literature Survey

The provided papers explore a spectrum of wireless charging technologies for electric vehicles, encompassing automatic initiation systems [1]. Proposes an automatic EV wireless charging system triggered by low battery, utilizing cloud data and service alerts for management [2]. Presents the design of a static IPT-based wireless EV charging system, discussing fundamental principles relevant to dynamic systems [3]. Proposes and validates a RIPT system for EV charging, detailing controller design and component sizing [4]. Reviews WPT technologies for EVs, emphasizing RIPT for automated and efficient charging infrastructure, with consideration for Coimbatore [5]. Offers an in-depth review of WPT for EVs, highlighting RIPT for automated and efficient charging infrastructure solutions relevant to Coimbatore [6]. Introduces a multisource system integrating solar roads and the grid for localized IR-triggered wireless charging of moving EVs [7]. Proposes and validates a short-range resonant inductive coupling system for wireless EV charging [8]. Explores dynamic wireless charging using Archimedean coils in lanes to extend EV range, with misalignment analysis [9]. Investigates the effect of EV speed on battery SOC in dynamic wireless charging using Matlab Simulink [10]. Proposes a sustainable EV charging method combining solar power generation with WPT technology [11]. Presents a framework for predicting charging power in static and dynamic wireless EV charging, considering coil parameters and speed, with renewable energy integration for regions like Coimbatore [12]. Discusses a survey comparing inductive power pad and resonant magnetic field coupling for wireless charging of moving EVs [13]. Explores on-road wireless EV charging as a complement to fast charging within smart grids, analyzing benefits for load management [14]. Reviews and classifies WPT systems for EV charging, focusing on inductive coupling, compensation, converters, control, and coil design [15]. Proposes and validates a DWC system for EVs on highways, integrated with PV units and battery storage. Reviews recent advancements in dynamic and quasi-dynamic wireless charging for e-mobility, highlighting potential and challenges. Discusses EV charging limitations and proposes V2V wireless charging as an innovative solution for en-route charging.

3 Proposed System

This proposed dynamic wireless charging project aims to revolutionize electric vehicle (EV) charging through a comprehensive two-system architecture encompassing both the power transfer (transmitter) and the energy capture (receiver) mechanisms. The receiver system, seamlessly integrated into the EV, comprises essential components such as GPS technology for precise location awareness, a receiver coil designed to capture transmitted energy, a voltage sensor for real-time monitoring of power levels, and a Wi-Fi-enabled controller to ensure smooth and reliable communication. Complementing this, the transmitter system is engineered for mobility within

the service team's vehicle and incorporates a high-frequency inverter to generate the necessary power for wireless transfer, a transmitter coil to facilitate this energy transmission, and a dedicated battery source to power the process. This detailed methodological breakdown provides a clear understanding of the development, practical application, and operational framework of this pioneering dynamic wireless charging system, which holds the promise of ushering in a new era of convenient and readily accessible EV power delivery within Coimbatore and beyond.

The overarching system architecture sown in Fig. 1 outlines a revolutionary EV charging initiative employing a dual strategy of GPS location tracking and wireless power transfer to significantly enhance charging convenience and accessibility. The system leverages a GPS module for accurate EV positioning and a voltage sensor to continuously monitor battery charge levels. Power for the transmitter can be sourced from an onboard battery or an external power supply, which, when utilized, feeds a high-frequency inverter that subsequently powers a wireless transmitter coil. On the receiving end, a wireless receiver coil integrated into the EV captures this transmitted power, which is then converted and regulated by a rectifier to effectively charge the vehicle's battery. A central power supply microcontroller serves as the intelligent core of the system, processing data from the GPS and voltage sensor and likely managing charging operations, while simultaneously maintaining communication with a cloud platform. Users requiring a charge can initiate a request via a service center portal, which activates GPS tracking and transmits the EV's precise location to a central web portal. This crucial information enables the efficient dispatch of mobile charging EVs (equipped with the transmitter systems) to wirelessly charge the requesting vehicle (featuring the receiver system). In parallel, the project envisions the strategic integration of wireless charging lanes into the existing infrastructure of Coimbatore. Approaching GPS-equipped EVs would automatically trigger the activation of built-in road transmitters, enabling wireless power transfer to the vehicle's receiver coil as it moves. The central service portal plays a vital role in managing user requests, monitoring the locations of both requesting and charging vehicles, and overseeing both the on-demand mobile charging and the infrastructure-based wireless charging services. The cloud platform acts as a central communication hub, facilitating seamless interaction among vehicles, the service center portal, and potentially a user-facing application, ultimately striving to significantly improve EV charging convenience and promote wider adoption of electric mobility within Coimbatore.

4 Algorithm and Flowchart

Step By Step Algorithm for the working of the project (Fig. 2):

1. System Startup: The electric vehicle's monitoring and charging logic begins execution.
2. Battery State Assessment: The current charge level of the vehicle's battery is continuously evaluated.

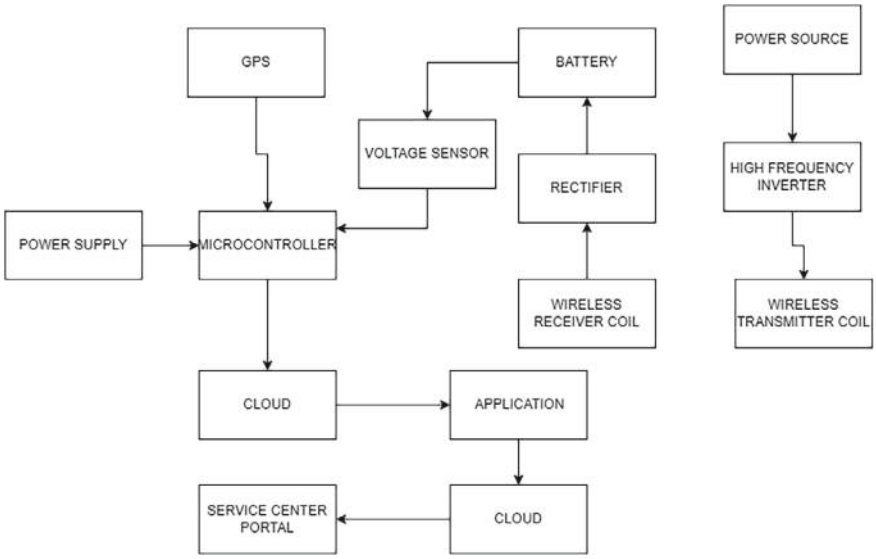


Fig. 1 Block diagram of the proposed system

3. Low Battery Condition Check: The system determines if the battery charge has fallen below a predefined critical threshold.
4. Cloud Data Transmission (Low Battery): If the battery is identified as drained, a notification, along with relevant data points, is transmitted to the cloud platform.
5. Charging Service Request Inquiry: The system checks if a request for charging assistance has been initiated, either automatically or by the user.
6. Service Portal Notification (Charging Needed): When a charging service request is active, a notification is dispatched to the designated service portal to facilitate the charging process.
7. Initiation of Wireless Charging: Upon confirmation of a service request, the wireless power transfer system is activated to begin charging the vehicle's battery.
8. Ongoing System Observation: The system continues to monitor the battery status and potentially other relevant parameters even after charging has commenced (as indicated by the feedback loop).
9. Periodic Parameter Reporting (Normal State): If the battery is not in a drained state, the system periodically sends relevant operational parameters (such as battery level) as part of its regular reporting.
10. Process Completion (Normal State): When the battery is not drained and the parameters have been transmitted, that particular cycle of monitoring and reporting concludes until the next assessment.

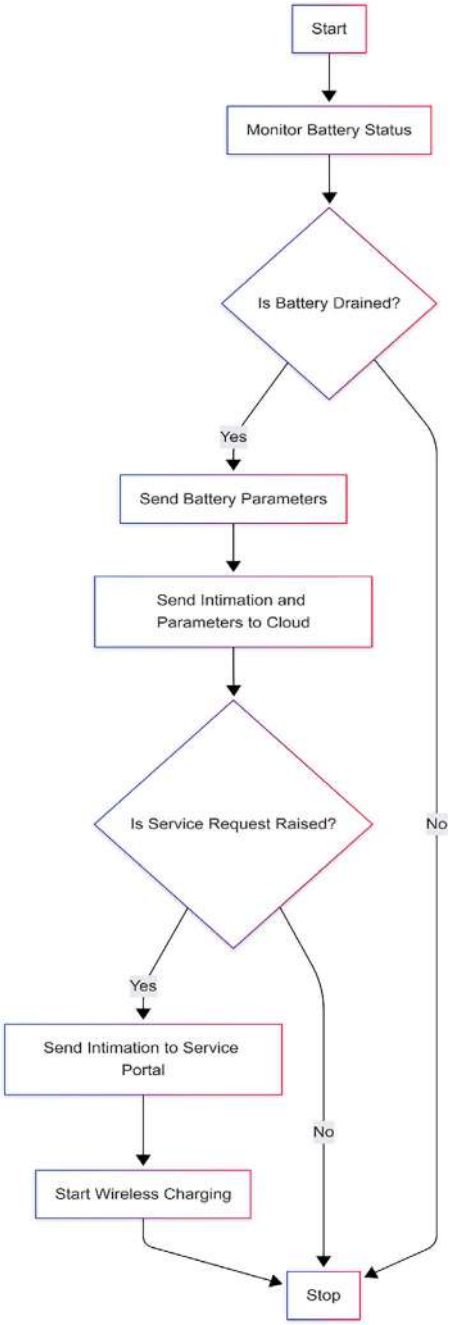


Fig. 2 Flowchart of the project

5 Simulation and Results

5.1 Hardware Output

The electric vehicle charging system prototype is shown in Fig. 3. The receiving vehicle is a small, four-wheeled platform on the left; it has a number of circuit boards (presumably for control, communication, and display), a yellow battery pack, an LCD screen, and a visible receiving coil. On a separate platform, the transmitting charging station is shown on the right. It has a transmitting coil, a larger black battery, and additional circuitry for power distribution and control. Because of their close proximity, the two coils show that the "electric car" and the "charging station" are being tested for inductive wireless power transfer.

Transmitting Kit

By using electromagnetic induction, the transmitting kit depicted in Fig. 4 is in charge of producing and sending wireless electricity to electric cars. It is made up of a transmitter coil, a high-frequency inverter, and a high-capacity battery. The necessary

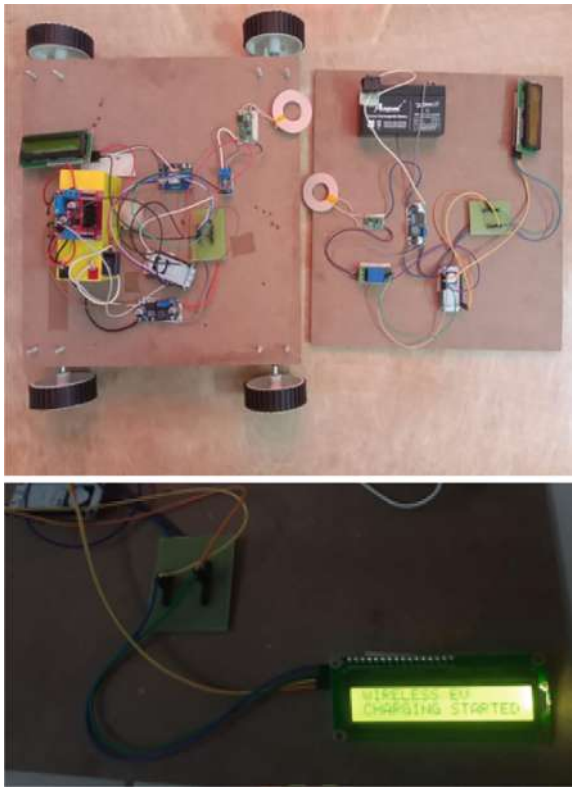


Fig. 3 Final output

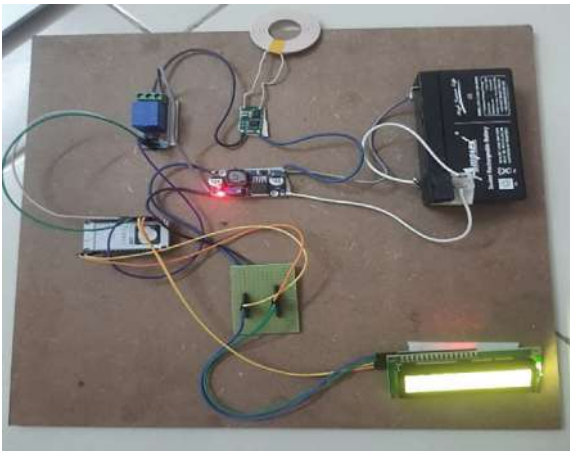


Fig. 4 Transmitting kit

energy is stored in the battery and transformed by the inverter into high-frequency AC electricity to produce an alternating magnetic field. This field is projected by the transmitter coil installed in the service team's car, enabling energy transfer to the receiver coil of the EV. For EV customers who require quick charging help, this configuration guarantees a mobile, on-demand charging system that lessens reliance on stationary charging stations and offers a flexible and effective power supply.

Receiving Kit

The reception kit shown in Fig. 5 is mounted inside the electric car, is made to absorb and transform the transmitter's wireless power into electrical energy that can be used. It has GPS technology, a receiver coil, a voltage sensor, and an integrated Wi-Fi controller. In order to generate an electrical current that is subsequently transformed into direct current (DC) for battery charging, the receiver coil absorbs the alternating magnetic field. While the Wi-Fi integrated controller enables smooth communication between the car and the service portal, the voltage sensor keeps an eye on battery levels to avoid overcharging or undercharging. As EVs get closer to approved charging stations, GPS tracking ensures that they receive power effectively, increasing accessibility and lowering range anxiety.

5.2 Software Output

The dynamic wireless charging network's software architecture serves as its backbone, facilitating smooth coordination and communication between service providers and electric vehicles. It has real-time tracking algorithms, a service site, and a mobile web application. Users can check the battery level of their car, request charging, and monitor service availability through the smartphone application. These requests are

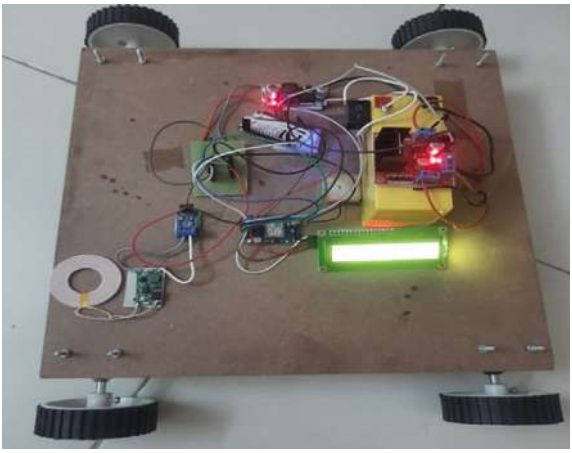


Fig. 5 Receiving kit

handled by the service portal, which assigns a nearby service truck with a transmitter to react as soon as possible. By continuously monitoring EV locations and dynamically modifying charging parameters for optimal energy transfer, real-time tracking algorithms guarantee effective power deployment. This software-driven method promotes the broad usage of electric vehicles, reduces manual intervention, and improves user convenience. Also, on successful completion of the charging of battery, the EV owners can make payment through the same webpage which has a button included in it. On clicking this button, the page gets redirected to any payment apps they have in their mobile phones. This ensures quick and easy payment for the charging done.

- *Design of Charging Request page on Receiver side*

Figure 6 shows the user flow for starting and finishing of vehicle-to-vehicle charging session is described in 6. When a customer clicks the "REQUEST FOR CHARGING" button on the receiver side page, they initiate the charging process. After that, the transmitter side receives the request. The customer can enter the OTP they received from the transmitter side and click "CHARGE NOW" to start charging. After the charging is complete, the customer can click the "Pay via Gpay" button to pay using any UPI app. The customer is taken to the payment app after clicking this button.

Figure 7 demonstrates a screen within an electric vehicle (EV) charging application, displaying a "Make Payment" interface. A listed entity titled "Available Charging Station" is presented, denoted as "EV Station 1," accompanied by a map option. At present, the "Nearest Charging Station" section remains vacant. A modal dialog situated at the bottom presents payment methods, including "Amazon Pay UPI" and "GPay". Also a "Do not ask again" option is provided for setting a default payment preferences.

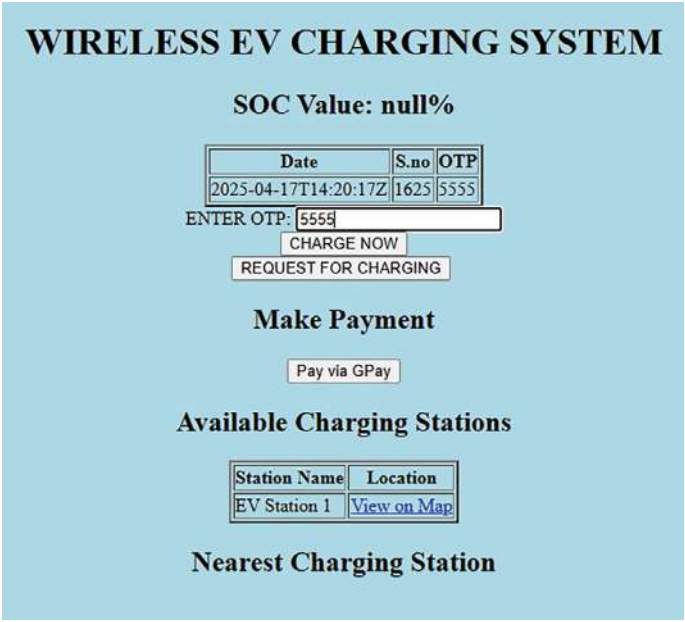


Fig. 6 Wireless EV charging APP interface with payment and charging station info

Fig.7 EV charging APP with access to GPAY or AMAZON pay

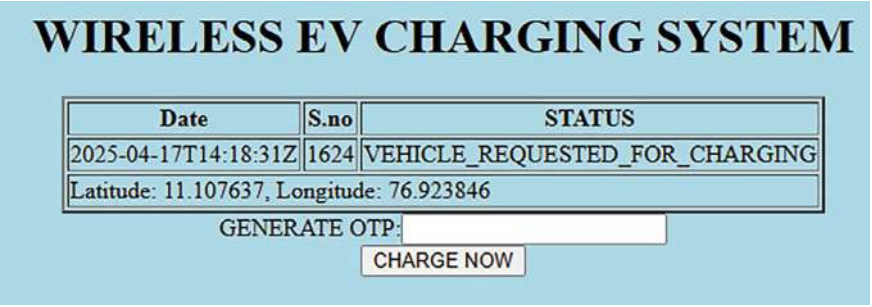


Fig. 8 EV charging request with vehicle location coordinates

- *Design of OTP generation page on Transmitter side*

Figure 8 demonstrates the interface of a wireless EV charging system, it shows that on April 17, 2025, at 6.31 P.M. IST, vehicle 1624 made a charging request. The system requires an OTP before the "CHARGE NOW" button can be used to start charging, as indicated by the "GENERATE OTP" button and input box. A pop-up for location access requests is also visible.

This is the transmitter side page on which the status is displayed as 'VEHICLE REQUESTED FOR CHARGING' once the client sends the request. The exact location i.e. the latitude and longitude of the customer vehicle is also shown on the page. On receiving the request, the server can generate an OTP by typing it on the input field box and click on 'CHARGE NOW'. Then the OTP is sent to the customer page.

5.3 Hardware Implementation

A. **NodeMCU (ESP8266)**

In the mobile charging devices and maybe in the wireless charging lanes, the NodeMCU (ESP8266) shown in Fig. 9 would probably act as the main control and communication device. Because of its integrated Wi-Fi, it's perfect for:

- Communication with the User Portal: In order to receive charging requests, send the location of the mobile unit, and maybe provide the user with updates on the charging status, it can connect to the specific user portal.
- GPS Data Processing: To precisely track the location of the mobile charging unit, the NodeMCU may receive and process the GPS data from the NEO-6M module.
- Control of Charging Operations: It may be able to regulate the voltage and current delivered to the requesting EV by programming it to control the power transfer process.

Fig. 9 NodeMCU (ESP8266)

B. **Voltage Detection Sensor Module**

In order to guarantee secure and effective power transfer, the Voltage Detection Sensor Module in Fig. 10 would be essential:

- Monitor the Voltage of Battery: The sensor checks the voltage of EV batteries in both the wireless charging lanes and mobile charging units to ensure that it stays within the allowed charging range.
- Prevent Overcharging or Undercharging: This system can guarantee required charging and also avoid overcharging by continuously monitoring the voltage, which would otherwise damage the battery.
- Feedback for Control Systems: During the charging process, the NodeMCU or other control circuits can use the voltage readings to modify the power output as necessary.

C. **NEO-6M GPS module**

The fundamental operation of the on-demand charging aid depends on Fig. 11 the NEO-6M GPS module:

- Accurate Mobile Unit Location Tracking: By giving the system precise GPS coordinates of the mobile charging units, this module enables the system to send the closest unit to the EV that is making the request.
- Location Reporting to the User Portal: EV owners will be able to track the location and anticipated arrival time of the mobile charging unit thanks to the transmission of GPS data to the user portal, most likely via the NodeMCU.

Fig. 10 Voltage detection sensor module

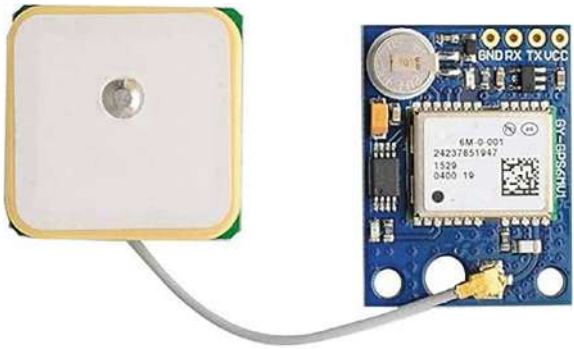


Fig. 11 NodeMCU (ESP8266)

- Direction to Wireless Charging Infrastructure: The driver can also be directed to the closest accessible wireless charging lane using the GPS module in the requesting EV.

D. **Rectifier**

One essential part of the wireless power transmission system is the rectifier:

- Converting AC to DC Power: Alternating current (AC) is usually sent during wireless power transfer. To transform this AC power into direct current (DC), which is needed to charge the EV's battery, a rectifier must be installed at the receiving end (in the EV or the mobile charging unit if receiving electricity wirelessly).
- Making Certain Compatibility: The rectifier makes sure that the EV's battery receives power in the proper DC format and voltage range for charging.

6 Conclusion

In summary, by integrating mobile transmitters in service trucks and receivers in electric vehicles (EVs), this novel wireless charging system tackles the issues of insufficient charging infrastructure and range anxiety. Utilizing technology like GPS and Wi-Fi, it offers a distinctive charging experience that improves driving range, reduces battery dependency, and expands accessibility in Coimbatore. Ongoing research and development is part of the future scope in order to standardize procedures, increase efficiency, and incorporate this infrastructure into Coimbatore's urban planning and transportation systems. This might establish Coimbatore as a leader in electric mobility innovation and make dynamic wireless charging a standard feature of EVs and public transportation, resulting in a more sustainable regional transportation system.

References

1. Khutwad, S.R., Gaur, S.: Wireless charging system for electric vehicle. In: 2016 International Conference on Signal Processing Communication Power and Embedded System (SCOPES), pp. 1281–1285 (2016)
2. Al Mamun, M.A., Istiak, M., Al Mamun, K.A., Rukaia, S.A.: Design and implementation of a wireless charging system for electric vehicles. In: 2020 IEEE Region 10 Symposium (TENSYMP), pp 1062–1065 (2020)
3. Lopes, P., Costa, P., Pinto, S.: Wireless power transfer system for electric vehicle charging. In: 2021 International Young Engineers Forum (YEF-ECE), pp. 13–18 (2021)
4. Shuguang, L., Jia, J.: Review of EVs wireless charging technology. In: 2019 IEEE 2nd International Conference on Electronics and Communication Engineering (ICECE), pp. 102–106 (2019)
5. Inductive wireless power transfer charging for electric vehicles—a review. IEEE Access **9**, 154957–154973 (2021)
6. Chandran, V., Ajisha, S., Ananthu, B., Devakrishnan, V., Pankaj, R.S., Akash, S.: Wireless charging of electric vehicles using solar road. In: 2022 International Conference on Innovations in Science and Technology for Sustainable Development (ICISTSD), pp. 1–6 (2022)
7. Khann-gern, W., Zenkner, H.: Wireless power charging on electric vehicles. In: 2014 International Electrical Engineering Congress (iEECON), pp. 1–4 (2014)
8. Wireless charging of electric vehicle while driving. IEEE Access **9**, 16493–16509 (2021)
9. Mohamed, N., Aymen, F., Mouna, B.: Wireless charging system for a mobile hybrid electric vehicle. In: 2018 International Symposium on Advanced Electrical and Communication Technologies (ISAECT), pp. 1–6 (2018)
10. Khan, K.L., Kant, R., Myneni, H., Bhat, A.H.: Wireless EV charging through a solar powered battery. In: 2022 1st International Conference on Sustainable Technology for Power and Energy Systems (STPES), pp. 1–5 (2022)
11. On-road wireless EV charging systems as a complementary to fast charging stations in smart grids. Sustainability (2024)
12. A review of wireless power transfer systems for electric vehicle battery charging with a focus on inductive coupling. Electronics (2022)
13. Dynamic wireless charging of electric vehicles using PV units in highways. Energies (2022)
14. A comprehensive review of the on-road wireless charging system for e-mobility applications. Front. Energy Res. (2022)
15. The problem of electric vehicle charging: state-of-the-art and an innovative solution. IEEE Trans. Intell. Transp. Syst. (2021)

Design and Implementation of a 5DOF Pick and Place Robotic Arm

Kukka Bharat, Ayush, Aayush, Vamshi, Monika Goyal, and Nitu Chauhan

Abstract This research study describes the design and implementation of a highly optimized 5-DOF robotic arm for cost-effectiveness, ease of use, and precise pick-and-place operations. The key feature of the robotic arm designed here was to be a very affordable and performance-oriented device using high-torque actuators for smooth, accurate joint motion. Its structure was designed on CAD software and fabricated in 3D printing, thus amenable to easy customization as well as lower production costs. By implementing a forward kinematics algorithm through the Denavit Hartenberg parameter method, the location and orientation of the end-effector can be accurately established. The graphical user interface developed is intuitive, allowing for control methods and automation. For serial communication of the control system, information communication is easier in order to ensure operation is smooth. This will hence be able to express a practical approach towards making robotic arms affordable, highly accurate, and user-friendly for multiple applications such as industrial automation and research platforms. With accessible fabrication techniques and robust control methods, the system presents a balanced solution for users who seek efficient and customizable robotic systems.

Keywords Robotic arm · Forward kinematics · DH parameters · GUI · Inverse kinematics · Serial communication

1 Introduction

The history of robotics started when, in the year 1954, George Devol invented the first industrial robot. Since then, development in robotics has been vast and wide-ranging. In the last few decades, it has achieved immense progress. Today, robots have become

K. Bharat (✉) · Ayush · Aayush · Vamshi · M. Goyal · N. Chauhan
Manav Rachna University, Faridabad, India
e-mail: bharathkukka86@gmail.com

N. Chauhan
e-mail: nituchauhan@mru.edu.in

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_9

unavoidable in many industries and transformed the industrial landscape of the world. From assembling parts to packaging, it is utilized on shop floors for the assembly of various components. In healthcare, surgical procedures demanding precision and complexity have been performed using robots. These impacts are obvious as robotics increasingly serves human activities in many areas, be it in its efficiency or safety.

Robotic arms especially replicate the dexterity and precision of the human arm, which makes them indispensable in repetitive tasks as well as hazardous ones. Designing robots for different applications varies considerably. While humanoid robots are designed to mimic human form and behavior and emphasize natural interactions, robots that are designed for healthcare have placed emphasis on precision, reliability, and safety in medical procedures.

In this paper, we discuss our findings on designing and developing a 5-DOF robotic arm with cost-effectiveness, ease of use, and precise control. The main goal of our research work is to design a strong and efficient robotic arm that will pick and place items with high precision and reliability.

2 Related Work

The paper [1] explores the simulation results of 1DOF clutched robotic arm; his work on a unique mechanical design with clutches where the arm makes use of just one motor to generate multiple 3D modes of motion. The proposed system makes use of a clutch-gearing mechanism that is either activatable or DE activatable, for various rotational motions. Recently there were new robotic arms in SOPHIA [2] and ATLAS [3]. They operate rather smoothly, flexibly, and almost human-like, but they are also pretty expensive and complicated systems not for any nonprofessional end-users.

Today, for humanoid and service robot's arms are still key parts, and they need to be more anthropomorphic, low energy consumption and safe. To these purposes, many researchers developed designs for anthropomorphic arms like, for example, in [4, 5].

In [6] discusses the development and implementation of a miniature robotic arm that can control a larger robotic arm with three degrees of freedom. In [7] author utilized accelerometers; the components of the acceleration parallel to the assumed axes (X, Y, Z) have relation among themselves with the total magnitude. Angles, velocity are obtained accurately.

In this [8] author discusses the design and path planning of a robotic arm designed for inspection of pipelines, more appropriately to pipeline inspection environments, requires detection of obstacles and avoiding obstacles while carrying out detailed inspections. IN [9] discusses about 2-axis and 3-axis robot manipulators, in [9] 4-axis, in [10] 5-axis, in [11] 6-axis.

In [12] author discusses the design and development of a 6DOF, known as PC-ROBOARM. Here, he considers the arm to be a three-link system wherein every joint is linked with a servomotor. The authors also introduce some software known

Table 1 Specifications of the motors

Joint	Actuator	Torque/step angle	Range	Voltage
Base	NEMA17 (metal gear)	4 kg cm/1.8	∓ 360	12 V [14.1]
Shoulder	NEMA17 (metal gear)	4 kg cm/1.8	− 3 to + 183	12 V [14.1]
Elbow	NEMA17 (metal gear)	4 kg cm/1.8	∓ 360	12 V [14.1]
Wrist	SG90 (plastic gear)	1.2 kg cm	0 to 180	3.0–7.2 V [14.2]
Gripper	SG90 (plastic gear)	1.2 kg cm	0 to 180	3.0–7.2 V [14.2]

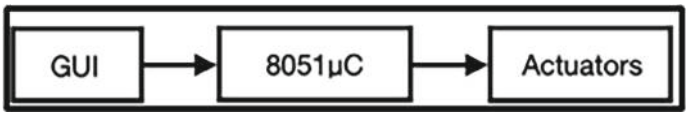


Fig. 1 Design flow of the robot

as SMART ARM(GUI) which helps in the design, simulation, and control of the robotic arm.

3 System Design and Architecture

This robot is having there NEMA17 Stepper motors at BASE, SHOULDER, ELBOW for higher torques and two SG90 servos at WRIST and GRIPPER for precise angular adjustments to facilitate controlled movements and precision across all joints. Designed a 3D CAD model of our robotic arm using SOLID WORKS [13] and we utilized 3D printer [14] to print all parts of the robot. In the control system, it will be the simplest and most cost-effective way by choosing the Aryabhata 805Micro-Controller. Microcontroller will get Commands from our Graphical user Interface (GUI) and performs the actions on the actuators. Table1 Shows the torque, range and operating voltages of the actuators used in this project.

In Fig. 1 the design flow of the robot is shown and then developed a Graphical user Interface using Python known as FlexArm5X. From this interface 8051µC will receive Commands using serial communication UART cable, and move the motors angles according to the commands received.

4 Methodology

"In mechanical systems study, it is always important to understand the motion of its different components. This is what defines kinematics: Kinematics is the study of motion of mechanical points, bodies and systems with due consideration of their

own physical properties as well as forces acting on them. This basic concept makes engineers and scientists analyze motion systematically and not incorporate displacement, velocity, and acceleration." Finding Position and orientation of the end effector at the given joint angles is known as Forward Kinematics [15] the common way to implement forward kinematic is by using Denavit-Hartenberg (DH) [16] parameterization a mathematical technique. first, we build a free body diagram of the robotic arm with this then we define coordinate frames at every joint, each frame is defined by four parameters known as DH parameters (link length (ai), link twist (αi), joint offset (di) and joint angle (θi). We will apply the Generic Link Coordinate Transformation Matrix from Base to End Effector. Figure 2 shows the Free Body diagram of the robot.

With help of MATLAB library DH table Solver [17] forward kinematic is solved for the robot, this library takes DH parameters link length (ai), link twist (αi), joint offset (di) and joint angle (θi) as input and solves all the homogeneous transformation matrix, this is a powerful tool for solving Forward kinematics, In Figs. 3 and 4 we can observe the homogeneous transformation matrix which represents position and orientation of the end effector with respect to base frame in home position.

The visualization of the robot is made using Robotic Tool Box MATLAB [18] and DH parameters created in order to verify the robot DH parameters, as shown in Fig. 5 we could see all the links and distance between links, axes of rotation at each joint.

After solving Inverse kinematic using Analytical method the resolved θ1, θ2, θ3 and θ4 are

$$
\begin{aligned}
zreach &= z - L1reach = \left(r^2 + Z^2 reach\right) \\
\theta 1 &= \arctan 2(y, x)\cos(\theta 3) = \left(reach^2 - L2^2 - L3^2\right)/2(L2L3) \\
\theta 3 &= \arccos(clip(\cos(\theta 3) - 1, 1)) \\
k1 &= L2 + L3\cos(\theta 3) \\
k2 &= L3\sin(\theta 3) \\
\theta 2 &= \arctan 2(Zreach, r) - \arctan 2(k2, k1) \\
\theta 4 &= 0
\end{aligned}
$$

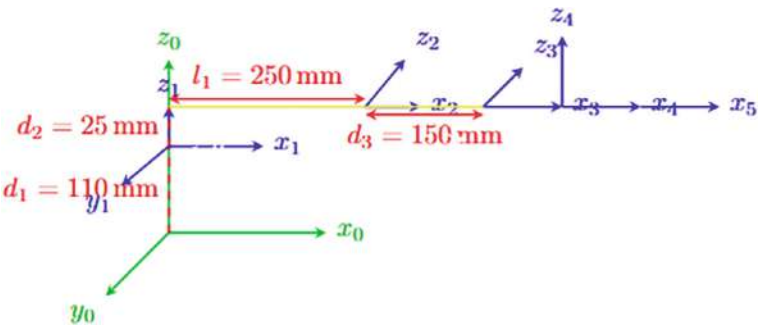


Fig. 2 Free body diagram of the 4DOF robot

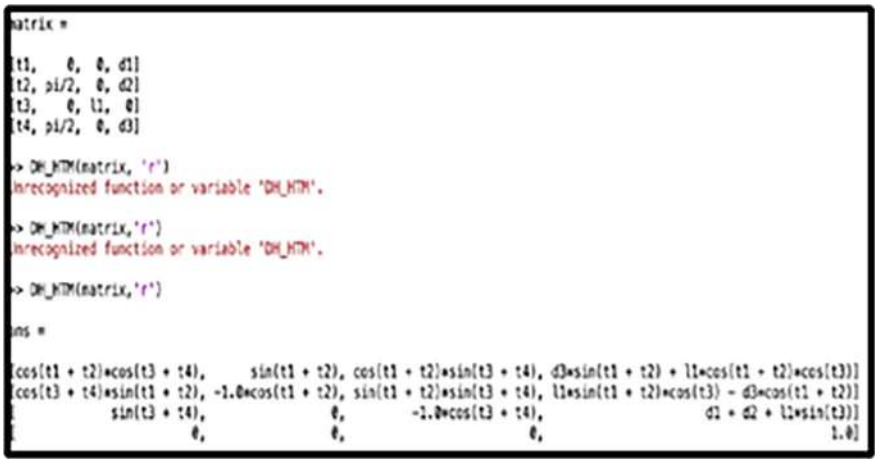


Fig. 3 Homogenous transformation matrix

Fig. 4 Position and orientation of the robot from base (home)

```
ans =

[1.0,   0,   0,           0]
[  0, 1.0,   0,          d3]
[  0,   0, 1.0, d1 + d2 + l1]
[  0,   0,   0,         1.0]
```

Set of all the Positions and orientations that the robot's end effector can reach comfortably (Reachable configuration of a robot) is known as Robots workspace [19] or a 3D volume of space that robots end effector can reach. The robot's workspace is dependent on the length of the links and joints. Reachable Workspace is the volume of space the end-effector can reach in at least one orientation. Dexterous Workspace. The Workspace that the end-effector can reach with all possible orientations it is smaller than the reachable Workspace. Our robots work calculates and graph using python's matplotlib library in Fig. 6 we can observe the workspace of the robot.

Whenever we work on any mechanical projects are being told before directly building the project first visualize the idea, this comes into life with the help of Software's like Free CAD, Solid Works etc.3D Modeling, simulation, motion analysis can be easily done by making use of these tools. In Fig. 7 we can observe the CAD design of the Robot, build using solid works.

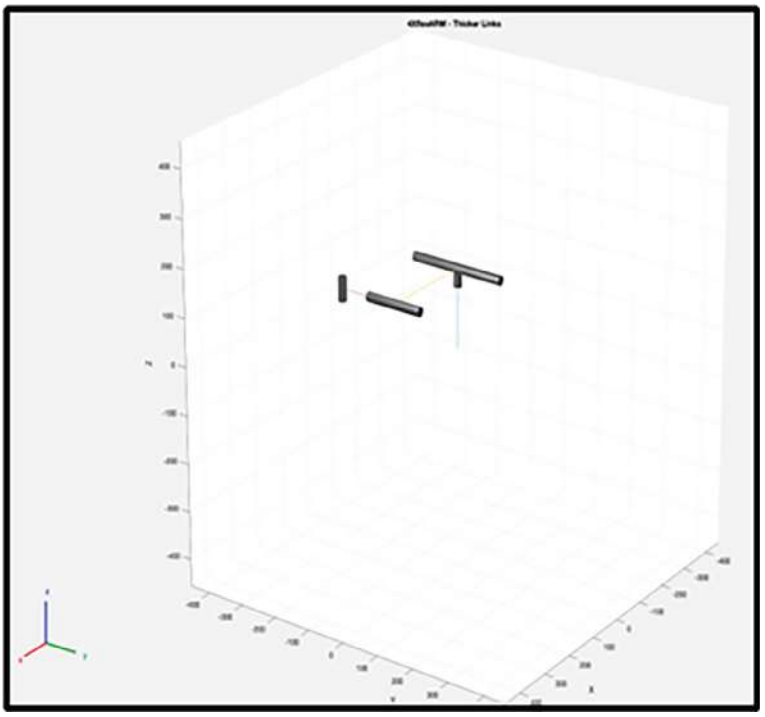


Fig. 5 Visualization of the robot

Fig. 6 Robots work space

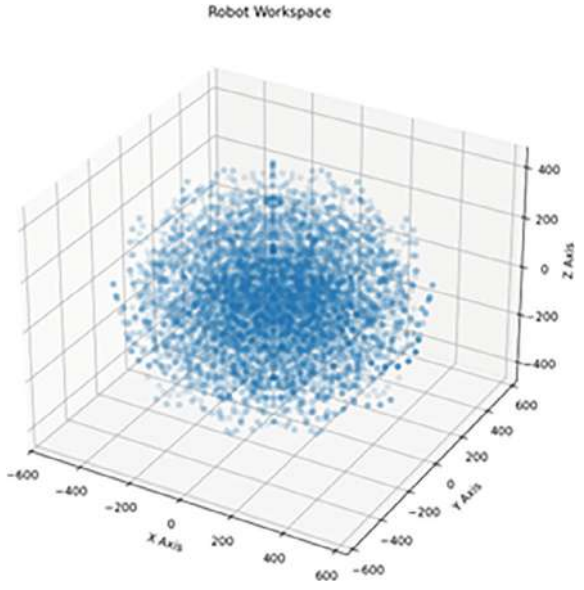
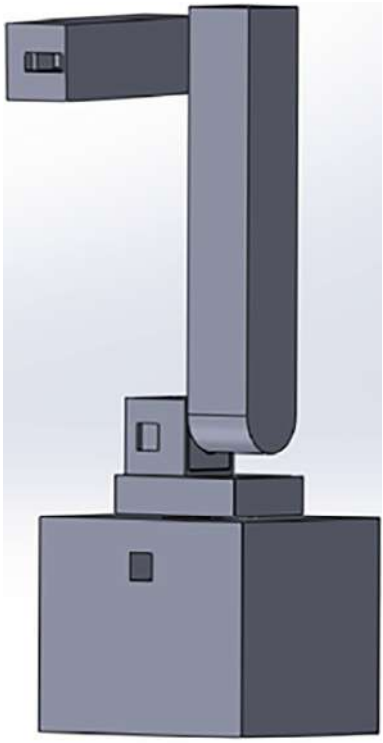


Fig. 7 CAD design



5 Interface

In Fig. 8, we have the Graphical user Interface of our project, with the help of UART cable we connect our PC to the 8051 MicroController Board. We have 5 sliders for 5 motors, when all motors are at 90° that is the robots home position. When we move sliders right or left accordingly the angle will change in the robot, the GUI with serial communication sends commands to the 8051μC (like M1089 mentions the motor and 089 denotes the angle) Microcontroller is coded to receive the commands and perform action, when 8051μC receives a command, it will look at the corresponding motor changes angle to step in case of stepper motors. In interface we have Home button to set all the robot home configuration and Record Play buttons to perform specific task continuously (example: when Record Pick is used then after whatever changes made in sliders will saved in a array, when Play Pick is used all the angles stored in the array are passed as commands to the 8051μC) this is a simple interface that is used to communicate between user and Robot with the help of PC (Fig. 9).

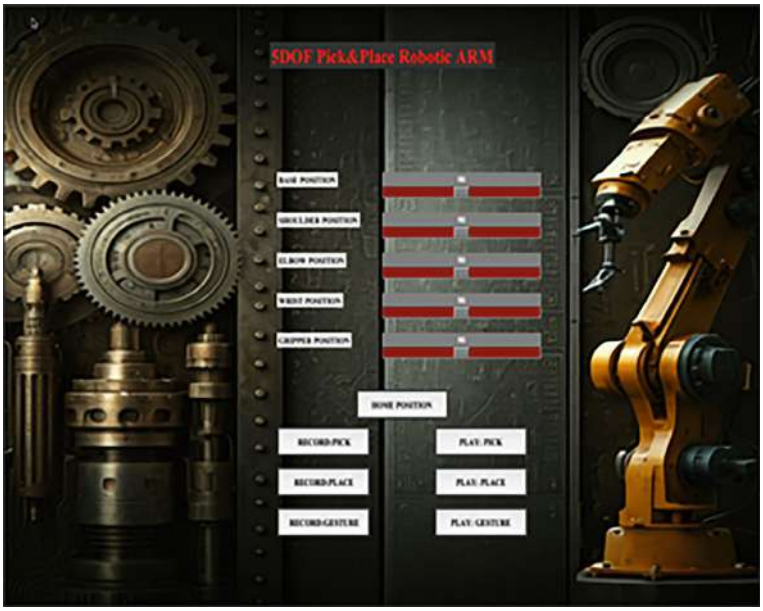


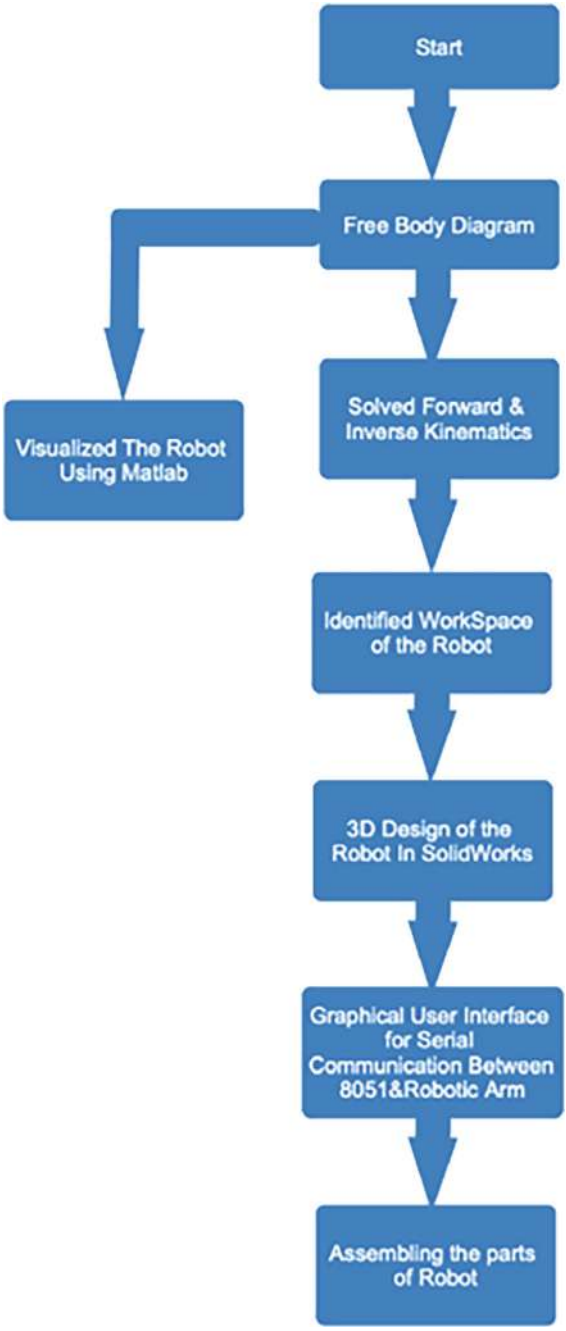
Fig. 8 GUI FlexArm5X

6 Results

The developed 5-DOF Robotic Arm able perform the action that is coming from the interface to the 8051 microcontrollers. Whenever angle is changed in the interface that is transferred to the microcontroller in the form of a command (like HOM, M1090, M2180, etc..) then Motors Rotates accordingly.

Variations in Angles as shown in Fig. 10, microcontroller programming involves the reception of commands through serial communication using PL2303 USB to TTL connector, whenever the command is received the microcontroller will calculates the required Steps and Direction for stepper motors and Pulse Width for servos and move to required position. Figure 11 shows the Robot position after performing the angle variations made in the FlexArm5X (GUI).

Fig. 9 GUI FlexArm5X



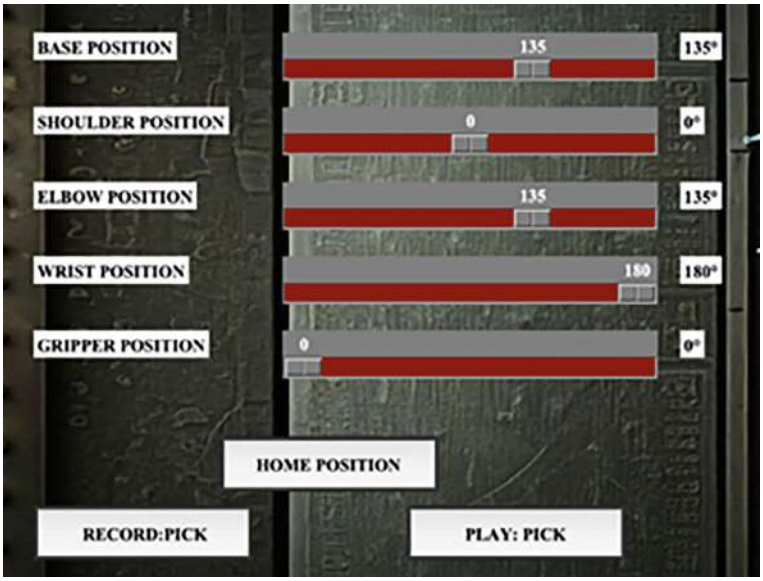
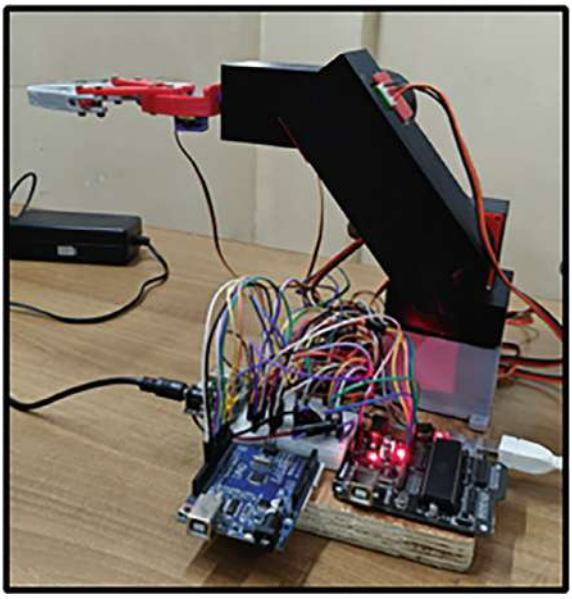


Fig. 10 Angle variations

Fig. 11 Robots position after motion



7 Conclusion

The implementation of the 5-DOF robotic arm for pick-and-place operations is working very precisely. Challenges such as cost efficiency, precision, and user-friendliness have been effectively addressed. By combining forward kinematics using Denavit Hartenberg (DH) parameters with a custom graphical user interface, the system has demonstrated reliable performance in pick-and-place tasks. The use of 3Dprinted components significantly contributes to affordability, while high-torque stepper motors ensure consistent and accurate movements. Through the integration of hardware and software, the robotic arm successfully full fills its intended purpose. This project highlights the potential of leveraging accessible fabrication techniques.

Future work: could focus on improving the arm's capabilities to make it even more user-friendly, Implementing Forward and Inverse Kinematics, trajectory Planning and Path Planning to make automations tasks.

References

1. Gu, H., Ceccarelli, M.: Simulation of combined motions for a 1-DOF clutched robotic arm. In: Proceedings of the 2009 IEEE International Conference on Mechatronics and Automation, pp. 3721–3729 (2009)
2. Hanson Robotics. https://www.hansonrobotics.com/sophia/
3. Boston Dynamics. https://bostondynamics.com/atlas/
4. Tuijthof, G.J.M., Herder, J.L.: Design, actuation and control of an anthropomorphic robot arm. Mech. Mach. Theory **35**, 945–962 (2000)
5. Iwata, H., Morita, T., Sugano, S.: Human symbiotic humanoid robot with whole-body compliance. In: Proceedings of the 14th CISM-!FToMM Symposium, Udine, Italy, pp. 537–547 (2002)
6. Megalingam, R.K., Boddupalli, S., Apuroop, K.G.S.: Robotic arm control through mimicking of miniature robotic arm. In: Proceedings of the 2017 International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 1–6 (2017)
7. Varghese, B., Thilagavathi, B.: Design and wireless control of anthropomorphic robotic arm. In: Innovations in Information, Embedded and Communication Systems (ICIIECS) Conference, pp. 1–4 (2015)
8. Electric Power Research Institute of Guangdong Power Grid & State Key Laboratory of Robotic Technology and System, Harbin Institute of Technology: In: Proceedings of 2020 IEEE International Conference on Mechatronics and Automation. IEEE (2020). https://doi.org/10.1109/ICMA49536.2020
9. Shirkhodaie, A., Taban, S., Soni, A.: AI assisted multi-arm robotics. In: 1987 IEEE International Conference on Robotics and Automation. Proceedings, vol. 4, pp. 1672–1676 (1987)
10. Huang, Y., Dong, L., Wang, X., Gao, F., Liu, Y., Minami, M., Asakura, T.: Development of a new type of machining robot—a new type of driving mechanism. In: 1997 IEEE International Conference on Intelligent Processing Systems, 1997. ICIPS'97, vol. 2, pp. 1256–1259. IEEE (1997)
11. Kuijing, Z., Pei, C., Haixia, M.: Basic pose control algorithm of 5-DOF hybrid robotic arm suitable for table tennis robot. In: 2010 29th Chinese Control Conference (CCC), pp. 3728–3733. IEEE (2010)

12. Bejo, A., Pora, W., Kunieda, H.: Development of a 6-axis robotic arm controller implemented on a low-cost microcontroller. In: 6th International Conference on Electrical Engineering/ Electronics, Computer, Telecommunications and Information Technology, 2009. ECTI-CON 2009, vol. 1, pp. 328–331. IEEE (2009)
13. Wong, G.H., Yap, Y.L., Lim, C.H.: 6-DOF PC-based robotic arm (PC-roboarm) with efficient trajectory planning and speed control. In: 2011 4th International Conference on Mechatronics (ICOM), pp. 17–19 (2011). https://doi.org/10.1109/ICOM.2011.5937116
14. SOLIDWORKS. https://www.solidworks.com/
15. SG90Servo. https://robocraze.com/blogs/post/
16. Koyuncu, B., Güzel, M.: Software Development for the Kinematic Analysis of a Lynx 6 Robot Arm. World Academy of Science, Engineering and Technology (2007)
17. DH. https://en.wikipedia.org/wiki/Denavit-Hartenberg_parameters; DH table solver https://in.mathworks.com/matlabcentral/fileexchange/103050-dh-table-solver
18. A robotics toolbox for MATLAB (1996). IEEE Journals & Magazine | IEEE Xplore. https://ieeexplore.ieee.org/document/486658
19. Workspace. https://www.sciencedirect.com/topics/engineering/robot-workspace

Enhancing Retail Insights: Introducing Dynamic Association Rule Mining over Deep Learning and Machine Learning

Abhay Nath, Aakanksha Kumari, Ruma Pal, Sachin Patel, and Amit Nayak

Abstract Enhancing retail decisions requires knowledge of how customers buy their products. Determining how customers group their purchases and anticipating more items they would buy constitutes a complex problem for retail companies. The research investigated ways to solve this problem by implementing a dynamic association rule mining algorithm which discovered valuable item relationships along with frequent itemsets. The interpretation of extracted patterns received enhancement through the utilization of a custom synergy score combined with zhang's score. The developed metrics generated more sophisticated understanding of item association insights which performed better than standard evaluation approaches. The pattern discovery results produced a significant enhancement because the highest synergy score reached 1.097 and the average score exceeded 1 which indicated more accurate and relevant mined rules. The proposed approach demonstrated superior performance than deep learning models which were used for comparison purposes. Analyzed findings enable businesses to use them for personalizing marketing strategies and inventory control and retail strategy development to support their data-driven decision-making while better serving client needs. The research presents an improved

A. Nath · A. Kumari · S. Patel (✉) · A. Nayak
Department of Information Technology, Devang Patel Institute of Advance Technology and Research, Charotar University of Science and Technology, CHARUSAT Campus, Anand, Gujarat, India
e-mail: sachinpatel.dit@charusat.ac.in

A. Nath
e-mail: 21dit040@charusat.edu.in

A. Kumari
e-mail: d23dit098@charusat.edu.in

A. Nayak
e-mail: amitnayak.it@charusat.ac.in

R. Pal
Faculty of Management Studies, Indukaka Ipcowala Institute of Management, Charotar University of Science and Technology, CHARUSAT Campus, Anand, Gujarat, India

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_10

method which analyzes buying patterns for customers while deriving meaningful business insights from retail information.

Keywords Retail analytics · Dynamic association rule mining · Synergy score metric · Indian retail dataset · Deep learning comparison · Customer purchase behavior

1 Introduction

From its beginning the retail sector established a record of dynamic adaptation to customer needs and industry developments [1]. The requirements of business necessitated genuine insights that allowed for better inventory optimization and customer satisfaction and profit maximization [2]. The analysis of retail sales data revealed that 90% went unused because there was limited insight potential [3]. Through data mining Walmart achieved supply chain excellence and Amazon generated 35% higher sales because of recommendation systems [4, 5]. Advanced analytical techniques for consumer behavior observation proved vital because they produced enhanced decision quality along with increased competitive position. To fulfill this requirement companies needed to develop new strategies which combined established and new technology platforms.

The global market suffers annual billons of dollars from under-utilized data which results in retail loss [6]. H&M experienced $4.3 billion worth of losses in 2018 because of inventory that did not sell [7]. The processing of modern retail data became difficult for traditional statistical analysis and basic clustering techniques [8]. The analytical methods of past times failed both in uncovering underlying patterns in data while also breaking down efficiently across large continuously changing datasets [9]. The limitations demonstrated that modern retail needs advanced methods for extracting valuable insights to support superior choices in retail operations.

Previous research studies explored different AI and machine learning approaches for retail analytics without achieving satisfactory outcomes. Research conducted on retail sales forecasting with XGBoost and LSTM shows an RMSE of 0.878 [10]. The error measurement remains moderately accurate although it fails to produce meaningful prediction capabilities. These models demonstrate difficulties in properly understanding retail data complexity according to the research findings. The identified performance deficit requires advanced forecasting techniques to reach improved results. Recognition techniques used in customer segmentation encounter difficulties achieving accurate results especially during conditions of class imbalance which results in unsatisfactory outcomes [11–13]. These approaches faced two major issues which made them unfit for real-world large-scale retail applications [14]. These outcomes established that retail analytics needed improved robust and efficient methods to achieve better results. In contrast to previous works that employed systematic datasets that offer simple patterns, our work is the first one to leverage a quite dynamic retail dataset, where pattern recognition is much harder

than for classical deep learning architectures. For comparison, we have done with the traditional method such as RNN, LSTM, Bi-LSTM and our proposed dynamic rule mining method outperforms all of traditional model to discover the complex and subtle rules in such environments.

The research objective focuses on using Association Rule Mining to study Indian retail store shopping behaviors to discover important interdependencies between commonly bought items. The research utilizes support metrics together with confidence and lift measurements alongside a proprietary Synergy Score to supply retailers with valuable business intelligence regarding their inventory control as well as their cross-sales methods and promotional planning initiatives. Association Rule Mining yields effective results when applied to dynamic retail datasets according to deep learning model comparisons.

The novel contribution of this paper:

- The study created its own dataset from genuine Indian retail store shopping data which enabled the study of realistic buyer behavior patterns.
- The method presents a dynamic association rule mining algorithm which controls support thresholds automatically through parallel processing while it discovers frequent itemsets.
- A new evaluation system merged Zhang's score and the custom synergy score to achieve improved data mining accuracy in extracted association rules.
- The proposed methodology achieved superior performance than deep learning models RNN, LSTM and Bi-LSTM while remaining the only models used for benchmarking.

2 Materials and Methods

In this study, the first step involved Indian retail dataset analysis and subsequent data preparation before performing dynamic apriori association rule mining. Research outcomes based on evaluation metrics and synergy scores, receive deep learning model comparisons for complete analysis (Fig. 1).

2.1 Dataset Overview

The dataset used in this study was created based on reports of shopping patterns in Indian retail stores. It included records of items frequently purchased together, such as 'Bread', 'Honey', 'Bacon', 'Toothpaste', 'Banana', 'Apple', 'Hazelnut', 'Cheese', 'Meat', 'Carrot', 'Cucumber', 'Onion', 'Milk', 'Butter', 'Shaving foam', 'Salt', 'Flour', 'Heavy cream', 'Egg', 'Olive', 'Shampoo', and 'Sugar'. The dataset highlighted purchase patterns, such as a tendency for customers purchasing salt and onions to also purchase eggs, based on historical trends [15].

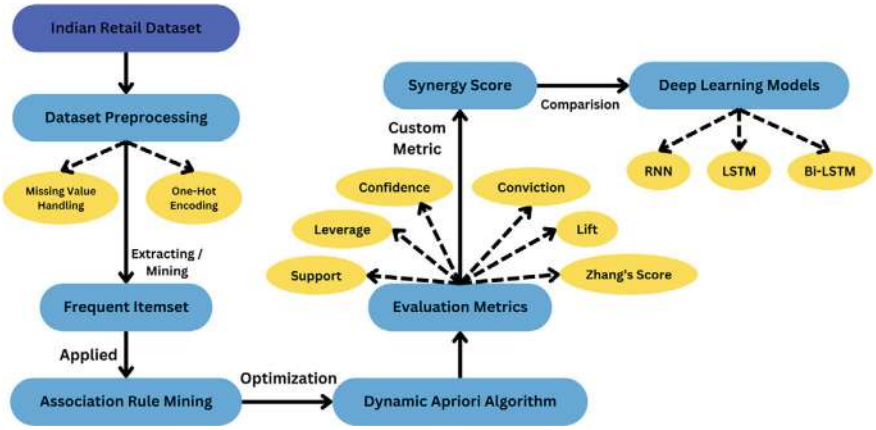


Fig. 1 The flowchart shows the sequence of steps which includes preprocessing the Indian retail data and running dynamic apriori rules together with result assessments using deep learning models for association mining

2.2 Dataset Preprocessing

A preprocessing step included various operations to fill zero values within missing data points then filter unneeded transactions to maintain relevant records followed by standard naming conventions across items. The data transactions were encoded as binary format through one-hot encoding technique [16]. A transaction appeared as a row while every column showed item purchase status as either (1) for buy or (0) for no buy. Frequent itemsets identification and meaningful association rule generation occurred after the dataset preparation process.

2.3 Frequent Itemset Mining

The apriori algorithm discovered all frequent itemsets within the dataset [17]. The algorithm discovers common item bundles through continuous item set addition until reaching a specific support requirement. Speaking of support detection determines the instance frequency of itemsets to reflect their practical value [18]. A particular calculation determines this value which can be seen in Eq. 1:

$$s(X) = \frac{Frequency\ of\ X}{Total\ Transactions} \tag{1}$$

For example, the combination of Salt and Onion appears in 40 transactions from a total of 100 so their support rate is 0.4. The support threshold value helps identify

important itemsets purchased with high frequency which are suitable for association rule generation.

The support threshold (s_{min}) adjusted automatically based on input dataset size to decrease unimportant combinations when working with small datasets but increase patterns for large datasets. The apriori algorithm determined candidate itemsets before removing those patterns with support levels lower than s_{min} and focused on important patterns through efficient computational management. The identified frequent itemsets served as the foundation to produce association rules.

2.4 Association Rule Mining

Association rule mining found relationships between items inside the dataset [19]. The structure of an association rule follows $X \rightarrow Y$ which describes the behavior of buying items from set X leading customers to purchase items in set Y [20]. Through this analytical approach retailers can detect purchasing patterns to make evidence-based strategic choices that enhance cross-promotional capabilities and maximum inventory distribution.

The confidence metric (c) provided the measurement system to evaluate rule reliability. The probability that Y purchase will occur under conditions of X serves as a measure called confidence and can be calculated according to Eq. 2 [21]:

$$c(\mathrm{X} \rightarrow \mathrm{Y}) = \frac{s(X \cup Y)}{s(X)} \tag{2}$$

The calculation uses support of X and Y jointly designated as $s(X \cup Y)$ alongside X's support value s(X). The analysis excluded weak associations by applying a threshold of 0.55 confidence because it helped identify strong patterns.

The lift metric (l) assisted in determining the importance of the rules [22]. Lift determines rule strength by examining the ratio between $X \rightarrow Y$'s observed association and its target independence value [23]. The method for lift calculation appears in Eq. 3:

$$l(\mathrm{X} \rightarrow \mathrm{Y}) = \frac{c(\mathrm{X} \rightarrow \mathrm{Y})}{s(Y)} \tag{3}$$

When lift exceeds 1, the variable X enhances Y's occurrence which signifies a positive correlation.

2.5 Algorithm Optimization

We optimized the apriori algorithm to achieve better performance and better scalability. The process of removing low-frequency items took place before itemset generation to avoid superfluous calculations. The minimum support value adjusted automatically according to dataset size for maintaining appropriate relationships with processing requirements. Parallelization of computations enabled the algorithm to process extensive datasets at higher speed for larger datasets. The improvements reduced computational time by finding relevant patterns at minimal resource costs within the process.

The Dynamic Association Rule Mining approach functions as a solution for eliminating both traditional association rule mining's static limitations and deep learning models' low capabilities with random and nonsequential retail data and inadequate evaluation measures. Dynamic support threshold adjustment combined with parallel operation and hybrid scoring allows the proposed approach to reach better accuracy rates while maintaining scalability and interpretability for decision support in India's dynamic market environment.

2.6 Evaluation Metrics

Multiple evaluation metrics analyzed the derived association rules to establish their quality standards for obtaining complete insights about their application worth. These evaluation metrics analyzed the item-to-item connection strength to produce results which were both practical and significant.

1. **Support**: The support evaluation metric provided information about itemset frequency by dividing transactions that include it against the total number of transactions. Support operated as a fundamental validation method to concentrate findings on established item patterns which grounded the rules with data evidence [18]. The formula shown in Eq. 1.
2. **Confidence**: The calculation for confidence measured likelihood by dividing transactions which include both antecedent elements against those with just antecedents. A proper assessment of rule reliability helps produce useful insights from associations identified through rules [21]. The formula shown in Eq. 2.
3. **Lift**: Lift calculates the relationship strength between antecedent and consequent items by dividing confidence by the support of the consequent item [24]. An association rule holds meaningful weight when its lift measure exceeds 1. The identification of meaningful and statistically significant patterns depended heavily on this particular measure [25]. The formula shown in Eq. 3.
4. **Leverage**: To calculate leverage the researchers used a statistical measurement which represented the independent value between observed patterns and calculated expected results when antecedents and consequents were statistically unrelated [26]. The logic system helped determine associations which transcended random chance relationships. The formula shown in Eq. 4.

$$Lev(\mathrm{X} \rightarrow \mathrm{Y}) = s(X \cup Y) - (s(X) \times s(Y)) \tag{4}$$

5. **Conviction**: The conviction value serves as a robustness indicator which shows how well the antecedent predicts the consequent absence. The rule's reliability received additional validation when conviction values exceeded 1 [27]. The formula shown in Eq. 5.

$$Conv(\mathrm{X} \rightarrow \mathrm{Y}) = \frac{1 - s(Y)}{1 - c(\mathrm{X} \rightarrow \mathrm{Y})} \tag{5}$$

6. **Zhang's Metric**: Zhang's metric served as an evaluation method which evaluated proportional antecedent-consequent differences while harmonizing between confidence levels and lift scores [28]. The approach proves valuable when finding worthwhile practical associations during analysis of unbalanced datasets. The formula shown in Eq. 6.

$$Zhang(X \rightarrow Y) = \frac{c(X \rightarrow Y) - s(Y)}{\max(s(X), s(Y)) - s(X) \times s(Y)} \tag{6}$$

2.7 Synergy Score: Custom Metric for Rule Evaluation

The Synergy Score (SS) allows users to find rules with high significance and reliability through its customized mathematical formula, shown in Eq. 7:

$$SS = c(\mathrm{X} \rightarrow \mathrm{Y}) \times l(\mathrm{X} \rightarrow \mathrm{Y}) \tag{7}$$

This metric increased the weight of highly confident rules that also have strong lifting capabilities to guarantee reliable selection for future analysis steps. Strong meaningful relations exist whenever the SS value surpasses 1.

3 Experimental Setup

A. **Hardware Environment**: The AI model underwent rigorous evaluation on a NVIDIA DGX STATION A100 computer, leveraging its advanced specifications to maximize performance potential [29]:

- Processor: Single AMD 7742, 64 Cores, operating at 2.25 GHz (Base)–3.4 GHz (Max Boost).
- System Memory: 512 GB DDR4.

- GPU: 4 NVIDIA A100 GPUs with 40 GB VRAM each.
- Performance: Capable of achieving 2.5 Peta FLOPS AI and 5 Peta OPS INT8.
- GPU Memory: 160 GB.
- Total System Storage: 1 × 1.92 TB NVME Drive.
- Internal Storage: 7.68 TB U.2 NVME Drive.
- System Network: Dual-port 10 Gbase-T Ethernet LAN and a Single-port 1 Gbase-T Ethernet BMC Management Port.
- Operating System: Ubuntu Linux.
- System Power Usage: Operates efficiently at 1.5 kW under 100–120 V_{ac}.
- Display: Features 4 GB GPU Memory and supports 4 × Mini DisplayPort.
- Operating Temperature: Maintained within 5–35 °C (41–95 °F).

B. **Software Environment**: The AI model was developed using Python as the primary programming language, with TensorFlow serving as the core machine learning framework. This comprehensive setup integrated a variety of software packages crucial for robust performance across diverse domains:

- Programming Language: Python.
- Machine Learning Framework: TensorFlow, Keras.
- Data Manipulation: Pandas, NumPy.
- Visualization: Matplotlib, Seaborn, Networkx.

4 Results and Discussions

4.1 Frequent Items Analysis

The analysis revealed the most commonly purchased items in the dataset based on their support values, directly indicating which products were frequently bought by customers. Support measures the frequency of an item in transactions, allowing sellers to understand customer purchasing habits. High-support items such as Banana (0.448), Cheese (0.444), and Bacon (0.431) highlighted their popularity among buyers. Other notable items included Hazelnut (0.420), Honey (0.416), and Heavy Cream (0.416), which were also frequently selected by customers. Staples like Bread (0.407) and Butter (0.375) further demonstrated consistent demand (Table 1).

These findings allowed sellers to identify patterns of sales and better understand item demand. For example, moderately frequent items like Milk (0.371), Shampoo (0.366), and Sugar (0.366) showed steady relevance. Additionally, essential products like Salt (0.399), Meat (0.388), and Floar (0.386) emphasized their role in retail transactions. By analyzing these support values, sellers could maintain inventory for high-demand items while exploring cross-selling strategies for related products, effectively leveraging purchasing patterns to optimize stock and improve customer satisfaction.

Table 1 Support values for the most frequently purchased items in the dataset

Item	Support	Item	Support
Banana	0.448	Salt	0.399
Cheese	0.444	Meat	0.388
Bacon	0.431	Floar	0.386
Hazelnut	0.420	Toothpaste	0.384
Honey	0.416	Olive	0.381
Heavy Cream	0.416	Cucumber	0.381
Carrot	0.414	Onion	0.379
Bread	0.407	Butter	0.375
Shaving Foam	0.405	Milk	0.371
Apple	0.405	Shampoo	0.366
Egg	0.403	Sugar	0.366

4.2 Top Association Rules with Synergy Score

The top association rules provided valuable insights into customer purchasing behavior, highlighting products often bought together. For instance, the rule (Onion, Bacon) → (Banana) indicated that if a customer purchased Onion and Bacon, there was a high likelihood they would also buy Banana. This rule achieved the highest Synergy Score (1.097), driven by its strong confidence (0.701) and lift (1.564). Similarly, the rule (Banana, Butter) → (Bacon) suggested that buyers of Banana and Butter were likely to purchase Bacon as well, supported by a confidence of 0.677 and a lift of 1.565 (Table 2).

Additional principal rules reinforced these associations between product concepts. The combination of (Olive and Shaving Foam) leading to Banana registered a Synergy

Table 2 Top association rules with corresponding evaluation metrics and synergy scores

Antecedents	Consequent	Confidence score	Lift score	Leverage score	Conviction score	Zhang's score	Synergy score
(Onion, Bacon)	(Banana)	0.701	1.564	0.047	1.846	0.444	1.097
(Banana, Butter)	(Bacon)	0.677	1.565	0.044	1.749	0.439	1.056
(Olive, Shaving Foam)	(Banana)	0.688	1.534	0.041	1.766	0.420	1.054
(Butter, Shaving Foam)	(Bacon)	0.670	1.555	0.045	1.727	0.441	1.043
(Bacon, Sugar)	(Meat)	0.632	1.629	0.046	1.664	0.476	1.030

Score of 1.054 while Butter with Shaving Foam resulting in Bacon produced a Synergy Score of 1.043. The identified item relationships helped sellers forecast customer tastes so they could develop successful promotional bundles and advertising tactics. Through these analytical findings retailers would offer improved shopping experiences by providing specific product bundles and achieve better inventory organization to boost sales.

4.3 Association Rule Network Analysis

The association rule network illustrated key relationships among items, with nodes representing products and edges indicating strong associations, shown in Fig. 2. Bacon emerged as a central node, linked to items like Cheese, Butter, and Banana, suggesting its significance in cross-selling. Banana also showed strong connections with Onion, Shaving Foam, and Olive, highlighting frequent co-purchases. Smaller clusters, such as Meat, Carrot, and Honey, revealed additional purchasing patterns. The connection between Salt and Heavy Cream, though subtle, indicated niche associations. This visualization provided valuable insights into customer behavior, complementing metric-based analysis and aiding in strategic product bundling and placement.

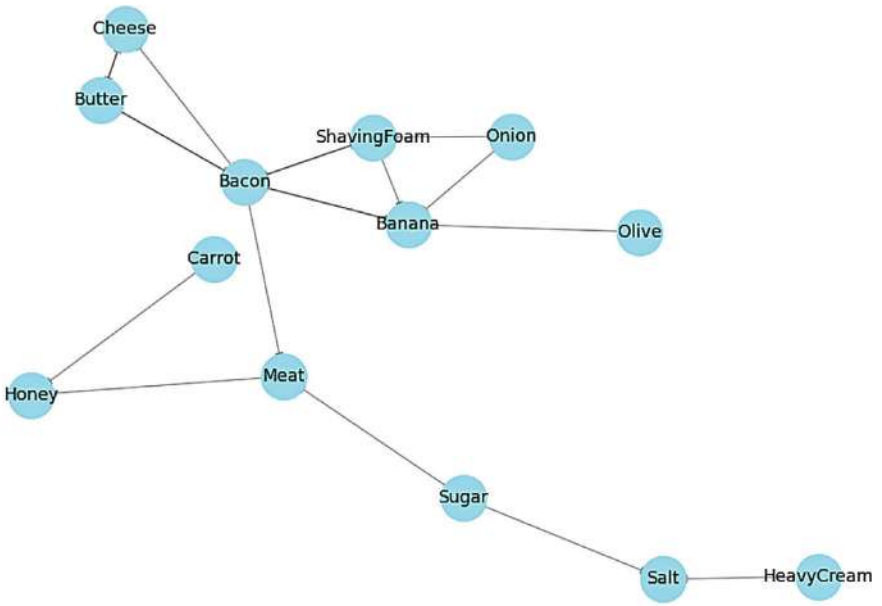


Fig. 2 Visualization of product associations, with nodes representing items and edges indicating significant relationships, highlighting key purchasing patterns and co-purchase trends

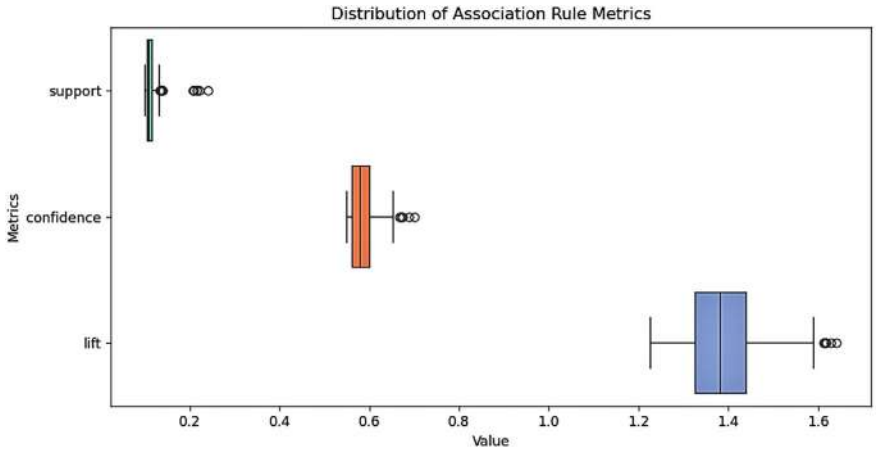


Fig. 3 Distribution of support, confidence, and lift metrics using boxplots for association rule evaluation

4.4 Metrics Distribution Analysis

The distribution of association rule metrics like support, confidence, and lift, highlighted varying contributions to rule evaluation, shown in Fig. 3. Support values were closely clustered, indicating consistent item frequencies in transactions. Confidence scores exhibited moderate variability, reflecting the reliability of predictions from antecedents to consequents. Lift values displayed a broader range, emphasizing their role in identifying stronger associations beyond random chance.

This analysis confirmed that lift effectively differentiated impactful rules, while confidence and support reinforced the reliability of item relationships. These metrics collectively provided a comprehensive evaluation framework for prioritizing meaningful association rules in retail datasets.

4.5 Confidence Versus Lift and Top Association Rules Insights

The hexbin plot visualized the relationship between confidence and lift for the association rules, highlighting the density of rules across various ranges, shown in Fig. 4a. Higher densities were observed in regions with moderate confidence (0.56–0.66) and lift values (1.3–1.5), indicating that rules in these ranges were more frequent and impactful. The gradient of the color scale revealed a few rules achieving both high confidence and lift, reflecting strong and meaningful relationships. This analysis validated the complementary roles of confidence and lift in identifying significant rules,

with lift providing insights into how much more likely a relationship was compared to random chance.

The radar plot depicted the top association rules by analyzing support, confidence, lift, and the Synergy Score. Each rule's metrics were plotted to highlight their relative importance, shown in Fig. 4b. For example, rules like (Onion, Bacon) → (Banana) and (Butter, Banana) → (Bacon) showed high confidence and lift, demonstrating their reliability. The Synergy Score amplified these patterns, identifying rules with strong combined significance. This visualization enabled a clear differentiation between rules and their contribution to the dataset, providing an insightful summary of impactful associations. Together, these analyses offered a comprehensive understanding of the most significant rules and their actionable insights.

4.6 Comparison with Deep Learning Models

The results clearly showed that commonly used deep learning models like RNN, LSTM, and Bi-LSTM struggled to perform well on our dataset, achieving only marginally acceptable validation accuracies (0.6147–0.6168) (Table 3) (Fig. 5) [30, 31]. This underperformance highlights their inability to learn patterns effectively from highly randomized and dynamic data. Despite extensive training, these models failed to capture meaningful relationships due to the complex and unpredictable nature of the dataset. In contrast, our dynamic Association Rule Mining technique demonstrated its superiority by effectively identifying significant patterns and relationships. This reinforces the robustness of our approach over traditional deep learning and machine learning techniques.

4.7 Comparative Discussion

Historical predictive methods fail to address the complex non-linear patterns with seasonality that exist within retail sales data. The LSTM network succeeds in predicting temporal patterns by resolving the gradient vanishing issue together with Random Forest Regression which effectively models non-linear variable interrelations. The methodologies have their primary application for identifying predictable patterns which exist in sales records of large shopping malls and retail chains that display systematic trends with consistent seasonality.

The dynamic association rule mining technique we have developed functions exclusively with datasets possessing high data dynamics which frequently occur in Indian retail sectors where pattern sequences remain uncommon. The method presents a dynamic support threshold control system which combines a synergy score with Zhang's metric while differing from previous research that used fixed support thresholds and standard metrics. The united approach strengthens predictive effectiveness along with dimensional scalability and produces interpretable rules

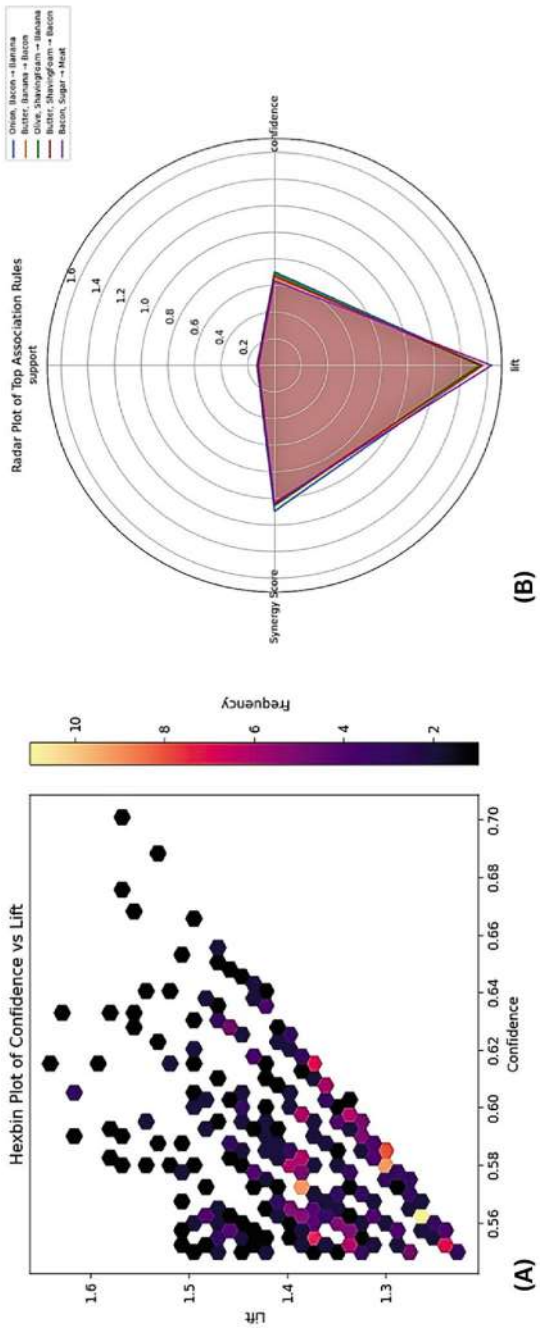


Fig. 4 Visualization of association rule metrics: **a** Hexbin plot showing confidence versus lift densities, **b** radar plot highlighting top rules by support, confidence, lift, and synergy score

Table 3 Comparison of RNN, LSTM, and Bi-LSTM models on learning rate, epoch, and accuracy

Model	Best learning rate	Best epoch	Training accuracy	Validation accuracy
Bi-LSTM	0.001	36	0.6038	0.6168
LSTM	0.001	3	0.5996	0.6159
RNN	0.001	48	0.6035	0.6147

which solve the complexities of discovering vital connections in unsequential retail information.

5 Conclusions

Research findings show that a dynamic association rule mining algorithm applied successfully to Indian retail data managed to optimize association rules through synergy score and zhang's score metric integration. The experimental findings showed that the algorithm obtained a superior capability in discovering meaningful item connections beyond the limits of support, confidence and lift measurement methods. Developing a novel dynamic association rule mining model constitutes the main contribution of this study because it achieves superior performance compared to deep learning techniques (RNN, LSTM, Bi-LSTM) while extracting meaningful patterns from Indian retail data. The method improved both extractable pattern interpretability and relevance to deal with fundamental problems in retail data analysis caused by missing values and uneven cases. Research findings revealed substantial prospects for how business decisions would benefit from personalization in marketing as well as stock management systems and strategic decision systems. This research faced two major constraints which included difficulties in processing greater datasets and difficulties in connecting to superior predictive models. Future investigations should adopt sophisticated deep learning frameworks to develop better pattern finding capabilities and optimize rule processing to establish real-time decision support platforms for retail analytics applications.

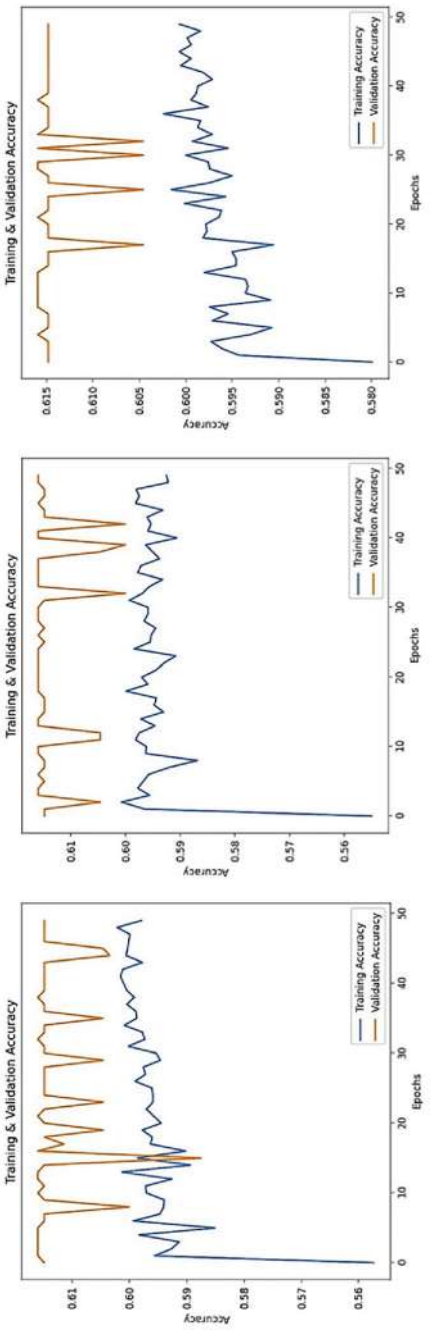


Fig. 5 Training and validation accuracy of **a** RNN, **b** LSTM, and **c** Bi-LSTM showing poor performance on highly randomized and dynamic data

Acknowledgements We extend our deepest appreciation to the Anita Devang Patel Ipcowala Center of Excellence in Artificial Intelligence, Charotar University of Science and Technology (CHARUSAT) for providing the NVIDIA DGX A100 Station, which was instrumental in our research. This advanced computing resource enabled us to conduct extensive experiments and achieve the significant results presented in this paper. We were profoundly grateful for their financial support to our research paper, which had been crucial to our project's success.

Data Availability The dataset used in this study is publicly available and can be accessed through the following GitHub repository link: https://github.com/AbhayNath001/Retail_with_ML/blob/main/market.csv. This dataset is provided in CSV format, and it contains all the relevant data used for the analysis in this research.

Declaration of Competing Interest The authors declare that they have no conflict of interests.

References

1. Pereira, M.M., Frazzon, E.M.: A data-driven approach to adaptive synchronization of demand and supply in omni-channel retail supply chains. Int. J. Inf. Manage. **57**, 102165 (2021). https://doi.org/10.1016/j.ijinfomgt.2020.102165
2. Niaz, M.: Revolutionizing inventory planning: harnessing digital supply data through digitization to optimize storage efficiency pre- and post-pandemic. Bullet J. Multidisip. Ilmu 1 (2022). https://journal.mediapublikasi.id/index.php/bullet/article/view/3534
3. Shaji George, A., Sujatha, V., Hovan George, A.S., Baskar, T.: Bringing light to dark data: a framework for unlocking hidden business value. Partners Univers. Int. Innov. J. **1**, 35–60 (2023). https://doi.org/10.5281/zenodo.8262384
4. Peesapati, S.R.Y.: Enhancing supply chain visibility in large enterprises: a literature review. Int. J. Supply Chain Manag. **13**, 20–27 (2024). https://doi.org/10.59160/ijscm.v13i3.6246
5. Karn, A.L., Karna, R.K., Kondamudi, B.R., Bagale, G., Pustokhin, D.A., Pustokhina, I.V., Sengan, S.: RETRACTED ARTICLE: Customer centric hybrid recommendation system for e-commerce applications by integrating hybrid sentiment analysis. Electron. Commer. Res. **23**, 279–314 (2023). https://doi.org/10.1007/s10660-022-09630-z
6. Ibrahim, A.E.A., Elamer, A.A., Ezat, A.N.: The convergence of big data and accounting: innovative research opportunities. Technol. Forecast. Soc. Change **173**, 121171 (2021). https://doi.org/10.1016/j.techfore.2021.121171
7. Kapić, B., Hosic, I.: The issue of fashion-waste towards the concept of circular economy and sustainability. In: 13th International Scientific Conference on Manufacturing Engineering Development and Modernization of the Manufacturing (2021). https://www.researchgate.net/publication/384675578
8. Formánek, T., Sokol, O.: Location effects: geo-spatial and socio-demographic determinants of sales dynamics in brick-and-mortar retail stores. J. Retail. Consum. Serv. **66**, 102902 (2022). https://doi.org/10.1016/j.jretconser.2021.102902
9. Zhou, S., Xu, H., Zheng, Z., Chen, J., Li, Z., Bu, J., Wu, J., Wang, X., Zhu, W., Ester, M.: A comprehensive survey on deep clustering: taxonomy, challenges, and future directions. ACM Comput. Surv. **57** (2024). https://doi.org/10.1145/3689036
10. Swami, D., Shah, A., Ray, S.K.B.: Predicting future sales of retail products using machine learning. ArXiv abs/2008.0 (2020). https://doi.org/10.48550/arXiv.2008.07779
11. Apichottanakul, A., Goto, M., Piewthongngam, K., Pathumnakul, S.: Customer behaviour analysis based on buying-data sparsity for multi-category products in pork industry: a hybrid approach. Cogent Eng. **8**, 1865598 (2021). https://doi.org/10.1080/23311916.2020.1865598
12. Nguyen, S.P.: Deep customer segmentation with applications to a Vietnamese supermarkets' data. Soft. Comput. **25**, 7785–7793 (2021). https://doi.org/10.1007/s00500-021-05796-0

13. Sokol, O., Holý, V.: The role of shopping mission in retail customer segmentation. Int. J. Mark. Res. **63**, 454–470 (2020). https://doi.org/10.1177/1470785320921011
14. Hassan, D.O., Hassan, B.A.: A comprehensive systematic review of machine learning in the retail industry: classifications, limitations, opportunities, and challenges. Neural Comput. Appl. (2024). https://doi.org/10.1007/s00521-024-10869-w
15. Nath, A., Pal, R.: Retail Marketing Data—North-East India (2025). https://doi.org/10.17632/5vr9vcwvrr.1
16. Dahouda, M.K., Joe, I.: A deep-learned embedding technique for categorical features encoding. IEEE Access **9**, 114381–114391 (2021). https://doi.org/10.1109/ACCESS.2021.3104357
17. Santoso, M.H.: Application of association rule method using apriori algorithm to find sales patterns case study of Indomaret Tanjung Anom. Brill. Res. Artif. Intell. **1**, 54–66 (2021). https://doi.org/10.47709/brilliance.v1i2.1228
18. Ramdhani, Y., Susanti, L.: The best association model on online retail datasets. In: 2024 International Conference on ICT for Smart Society (ICISS), pp. 1–6 (2024). https://doi.org/10.1109/ICISS62896.2024.10751212
19. Wu, W.-T., Li, Y.-J., Feng, A.-Z., Li, L., Huang, T., Xu, A.-D., Lyu, J.: Data mining in clinical big data: the frequently used databases, steps, and methodological models. Mil. Med. Res. **8**, 44 (2021). https://doi.org/10.1186/s40779-021-00338-z
20. Alam, T.M., Shaukat, K., Hameed, I.A., Khan, W.A., Sarwar, M.U., Iqbal, F., Luo, S.: A novel framework for prognostic factors identification of malignant mesothelioma through association rule mining. Biomed. Signal Process. Control **68**, 102726 (2021). https://doi.org/10.1016/j.bspc.2021.102726
21. Jia, X., Zhang, D.: Prediction of maritime logistics service risks applying soft set based association rule: an early warning model. Reliab. Eng. Syst. Saf. **207**, 107339 (2021). https://doi.org/10.1016/j.ress.2020.107339
22. Fister, I., Fister, I.: Information cartography in association rule mining. IEEE Trans. Emerg. Top. Comput. Intell. **6**, 660–676 (2022). https://doi.org/10.1109/TETCI.2021.3074919
23. Rella Riccardi, M., Mauriello, F., Scarano, A., Montella, A.: Analysis of contributory factors of fatal pedestrian crashes by mixed logit model and association rules. Int. J. Inj. Contr. Saf. Promot. **30**, 195–209 (2023). https://doi.org/10.1080/17457300.2022.2116647
24. Bao, F., Mao, L., Zhu, Y., Xiao, C., Xu, C.: An improved evaluation methodology for mining association rules. Axioms **11** (2022). https://doi.org/10.3390/axioms11010017
25. Moreno Ribera, A.: Association Rules for Predictive Purposes Applied to Omics Data. Complutense University of Madrid (2024). https://hdl.handle.net/20.500.14352/109010
26. Shinde, V.: Prediction of Co-occurrence of Antimicrobial Resistant (AMR) Genes in Salmonella and Enterococcus Using Bayesian Networks and Association Rule Mining. University of Georgia (2022). https://www.proquest.com/openview/eca33b1a4831e0147fb82a7bc4eff07e/1
27. Darrab, S., Broneske, D., Saake, G.: Exploring the predictive factors of heart disease using rare association rule mining. Sci. Rep. **14**, 18178 (2024). https://doi.org/10.1038/s41598-024-69071-6
28. Tao, J., Bai, W., Peng, R., Wu, Z.: Sustainable regional straw utilization: collaborative approaches and network optimization. Sustainability **16** (2024). https://doi.org/10.3390/su16041557
29. Nath, A., Raval, C.U.: Transforming Indian digital landscapes: a study on generative AI-powered voice assistants. In: 2024 IEEE 3rd World Conference on Applied Intelligence and Computing, pp. 700–704 (2024). https://doi.org/10.1109/AIC61668.2024.10730939
30. Sherstinsky, A.: Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Phys. D Nonlinear Phenom. **404**, 132306 (2020). https://doi.org/10.1016/j.physd.2019.132306
31. da Silva, D.G., de M. Meneses, A.A.: Comparing long short-term memory (LSTM) and bidirectional LSTM deep neural networks for power consumption prediction. Energy Rep. **10**, 3315–3334 (2023). https://doi.org/10.1016/j.egyr.2023.09.175

Synthetic Data Factory: Scalable and Domain-Agnostic Data Generation with Generative AI and Statistical Fidelity

Golagabathula Jyothi, M. Varshith Rao, M. Lahari Priya, Jay Patel, and T. Varun Kalyan

Abstract Synthetic data serves as an innovative solution that tackles major problems which include privacy concerns and limited availability and unbalanced data resources. The presented pipeline "Synthetic Data Factory: Scalable and Domain-Agnostic Data Generation with Generative AI and Statistical Fidelity" delivers an extensive data generation solution. A machine learning system produces synthetic data of high quality in different domains through its implementation of CTGAN and Gaussian Copulas. The designed framework maintains identical statistical characteristics to real data by protecting practical value so it passes ML model examinations. The data generation system integrates four essential elements which consist of data preparation, synthetic data production and statistical assessment and machine learning-based utility testing. This pipeline provides scalability alongside domain adaptability along with matched performance between synthetic and original data which signifies its potential as an AI application privacy solution.

Keywords Synthetic data · Generative models · LLMs · GANs · Data privacy · Pipeline · Data validation · AI model training · Differential privacy · Data visualization

1 Introduction

The fast-paced digital transformation era established data as the foundation for AI innovation and ML developments. Machine learning models achieve effective results based on the availability of rich datasets that reflect all spectrum groups and maintain high data quality. The acquisition of such datasets from actual sources proves challenging for many organizations. The regulations of GDPR and HIPAA limit

G. Jyothi (✉) · M. Varshith Rao · M. Lahari Priya · J. Patel · T. Varun Kalyan
Sreyas Institute of Engineering and Technology, Hyderabad, India
e-mail: jyothi@sreyas.ac.in

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_11

staff members from accessing sensitive data but data retrieval often requires substantial time and funds and is specific to particular industries. Real data suffers from multiple problems that include class imbalance together with missing values and sparsity which harm model performance and fairness metrics. The solution to these constraints has brought forward synthetic data generation as an essential method. Synthetic data serves as carefully designed computer-generated data which duplicates the features of authentic information but keeps actual user or patient details hidden. The information remains freely distributable while production procedures generate abundant data to fit unique use requirements. The process of creating synthetic data enables opportunities to build machine learning models and validate them across restricted areas such as healthcare and retail sectors and finance sectors.

The Synthetic Data Factory represents a domain-independent scalable framework that produces high-quality synthetic datasets through Generative AI implementations of CTGAN (Conditional Tabular GAN) and Gaussian Copula-based methods [1]. The pipeline maintains both statistical accuracy and practical value of the created data. The pipeline contains different interconnected components starting from real data ingestion to preprocessing and domain configuration and synthetic generation and statistical comparison and ML model evaluation.

An essential feature of this system originates from its domain-independent and entirely modular design structure. The system modifies structured data inputs from any domain while sustaining domain-related semantic rules (e.g. healthcare age restrictions and financial transaction categories). The statistical fidelity module depends on Jensen-Shannon Divergence and correlation matrices to confirm the alignment between synthetic and real data. A comparison of classifier performance takes place during utility evaluation when development includes training of Decision Trees Random Forests Logistic Regression and XGBoost classifiers on real and synthetic datasets.

Through its pipelines developers gain access to synthetic data which simultaneously meets privacy standards while being both statistically valid and usable for ML processes. The method shows promise for secure model development tests as well as data improvement methods while enabling synthetic AI performance assessments and fairness testing routines. The project advances responsible AI research by resolving issues that exist between available data and data privacy protection.

Organizations gain two key benefits from synthetic data because it both broadens access to important information while protecting privacy and it enables secure testing and experimentation that prevents legal or ethical issues. Organizations need this approach in particular for regulated sectors since they maintain strict rules about sensitive information. Synthetic data provides organizations with adaptive capabilities to generate challenging data formats or well-balanced datasets which would otherwise be hard to obtain from existing real-world systems. This project combines advanced generative AI techniques to create synthetic data which duplicates original dataset characteristics while preserving realistic consistency across different domains. The Synthetic Data Factory helps companies bridge data shortages and model achievements to support industrial innovation and ethical AI development.

This paper's primary contribution is the creation of an extensive framework for the generation and assessment of synthetic data that is independent of domain. This pipeline produces high-quality synthetic datasets by utilizing sophisticated generative AI models, specifically CTGAN and Gaussian Copulas that mimic the statistical and structural properties of actual data. To standardize inputs prior to model training, the framework incorporates a versatile preprocessing module that handles missing values, encodes categorical variables, and scales numerical features. Crucially, it facilitates domain-specific configuration, enabling customization for industries like retail, healthcare, and finance. The dual-mode evaluation approach is what distinguishes this work: a utility evaluation module benchmarks predictive performance using machine learning models (Decision Tree, etc.), while a statistical fidelity module compares real and synthetic data distributions using metrics like Jensen-Shannon Divergence and KS test.

2 Literature Survey

Synthetic data generation has gained significant attention due to increasing privacy concerns and the demand for data availability. Several methods have been proposed across domains, particularly leveraging generative models such as GANs and statistical copula-based techniques. Choi et al. [2] introduced MedGAN, a GAN-based approach to generate realistic healthcare records, which inspired further domain-specific synthetic data models. However, MedGAN lacked generalizability across domains and was limited in handling mixed data types. CTGAN [3] extended traditional GANs by effectively modeling tabular data with high cardinality categorical variables. While effective, CTGAN alone does not ensure statistical similarity or assess downstream utility of the generated data, which our approach addresses. On the statistical front, Gaussian Copulas have been used to model multivariate dependencies [4], but without integrated domain control or utility validation. Recent works such as Yoon et al. [5] in **SDV** emphasize fidelity but are often evaluated only through distributional metrics without rigorous machine learning utility testing. Moreover, prior pipelines often lack domain-specific configuration or integration of preprocessing, statistical fidelity, and ML-based evaluation into a single modular system.

S. No.	Title	Author(s) and year	Algorithms/ methods	Description
1	Synthetic Data Generation for Machine Learning: A Review	Patki et al. [4]	GANs, CTGA Bayesian Networks	Reviews synthetic data methods and compares utility and privacy trade offs

(continued)

(continued)

S. No.	Title	Author(s) and year	Algorithms/ methods	Description
2	Deep Learning for Medical Image Analysis: A Comprehensive Review	Xu et al. [3]	CTGAN, Mode specific Normalization	Proposes CTGAN to handle mixed data types more effectively
3	Evaluating Synthetic Data Utility and Privacy in Tabular Datasets	Jordon et al. [6]	XGBoost, C2ST, Decision Trees	Introduces a framework to evaluate data utility and privacy
4	SynthEval: A Framework for Benchmarking Synthetic Data	Yoon et al. [5]	Random Forests, Logistic Regression	Presents a toolkit to benchmark synthetic data quality
5	Fairness and Utility in Synthetic Data Generation	Patki et al. [4]	FairGAN, Bias Metrics	Focuses on fairness aware synthetic data generation

3 Problem Statement

Machine learning coupled with data-driven decision-making processes has caused high-quality data needs to skyrocket over recent years. Large-scale acquisition of diverse labeled data that meets privacy requirements poses a fundamental challenge which mostly affects healthcare and financial as well as retail settings. Real-world datasets include missing values together with class imbalances and personal information which reduces their potential for use by researchers. The service restrictions create obstacles that limit both the development of versatile machine learning models and speed up innovation and raise ethical and legal troubles.

The standard methods used to anonymize data prove either inadequate or damage valuable statistical metrics making it impossible to ensure absolute data privacy protection. A flexible domain-independent system should be developed immediately because it needs to generate authentic synthetic data with accurate statistics and function well for machine learning applications.

The development of a Synthetic Data Factory represents the core objective of this project to build a pipeline which utilizes Generative AI models like CTGAN and Gaussian Copulas to produce statistical match data that lacks recognizable information. Alongside preprocessing it integrates evaluation and visualization modules that enable the generated data to fulfill statistical requirements and practical standards while providing industries with safer and more efficient AI model development capabilities.

4 Objectives

The main goal of this work is to build a flexible data generation system that produces synthetic data which accurately represents authentic dataset characteristics across domains. The created artificial data must maintain the statistical properties of actual information while providing models a sufficient framework to conduct training and testing alongside equivalent performance results. An AI system applies CTGAN and Gaussian Copulas generative models to solve data deficit problems while addressing privacy issues as well as regulatory conditions in the system. Through its integrated processing frame work the system performs statistical verification and practical assessment of synthetic data to guarantee its statistical effectiveness and usefulness in downstream AI applications.

1. Create a generic synthetic data production framework which operates with interchangeable modules for different domains.
2. The statistical characteristics of genuine datasets remain intact through deployment of CTGAN and Gaussian Copulas models.
3. The system implements configurable rules to process and generate synthetic data which conforms to specific domain conditions.
4. The statistical reliability of synthetic data should be assessed through divergence metrics together with correlation similarity.
5. An evaluation of data utility should involve training Decision Tree, RF, Logistic Regression, XGBoost models which allows comparison of results obtained from synthetic and real datasets.
6. A system must allow users to interpret results by showing metric comparisons as well as plots.

5 System Architecture

A modular pipeline system architecture enables the analysis and synthetic data evaluation process across different domains for multiple setups. Each part of the architecture system helps create data with enhanced quality standards and higher fidelity through improved utility performance. This design allows for repeated use as well as data expansion and straightforward data interpretation which suits a wide range of datasets and specialized business sectors (Fig. 1).

1. Data Ingestion:
   (a) The system retrieves actual datasets from medicine departments and bank industries and retail sectors.
   (b) The system operates with uniform data formats through its domain configuration abilities.

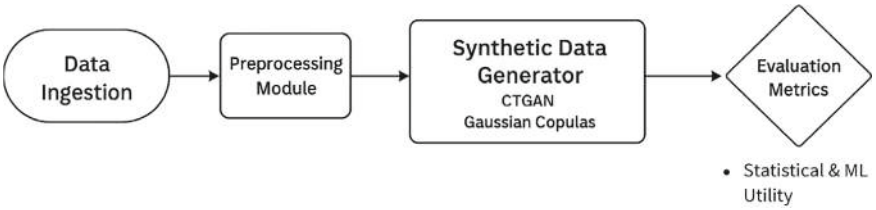


Fig. 1 System architecture

2. Preprocessing Module:
   (a) This module deals with multiple data pre-processing tasks which include addressing missing values and tallying values as well as implementation of scaling techniques and outliers treatment.
   (c) This module configures the data for both synthetic generation tasks and ML assessment needs.
   (d) Supports domain-specific preprocessing logic.
3. Synthetic Data Generator
   (a) CTGAN serves as the tool for synthesizing data when facing highdimensional along with imbalanced datasets.
   (b) The data processing utilizes Gaussian Copulas as a statistical tool for maintaining original property preservation.
   (c) The tool replicates original data patterns when creating synthetic datasets.
4. Evaluation Metrics:
   (a) Statistical Metrics: JS Divergence, Correlation similarity.
   (b) DT, RF, LR, XGBoost models demonstrated the following ML utility metrics: Accuracy alongside Precision and Recall along with F1-Score.
   (c) The evaluation assesses how matching model results perform when evaluated between actual datasets and synthetic data sets.

6 Methodology

6.1 Proposed Work

Research undertakes the development of a flexible domain-independent framework to produce synthetic data solutions which retain statistical validity alongside practical use. The solution creates high-quality synthetic datasets to replace real data in machine learning tasks while resolving data privacy and availability issues together with imbalance problems that organizations confront.

The system architecture functions through distinct modular components which include real data ingestion along with preprocessing steps and a synthetic data generation unit operated by CTGAN and Gaussian Copulas models accompanied by evaluation modules for statistical and performance assessments.

Datasets extracted from healthcare and financial and retail sectors undergo detailed preprocessing operations prior to training which includes data encoding along with normalization and missing data imputation techniques. Synthetic data generators receive ready data from preprocessing that trains them properly.

There are two fundamental methods to evaluate the synthetic datasets.

- The generated data passes statistical evaluation by using Jensen-Shannon Divergence alongside Correlation Matrix similarity to verify distribution fidelity.
- Two types of evaluation occur in the Machine Learning Utility Evaluation stage. This stage includes training distinct ML models from Decision Trees to XGBoost to Logistic Regression to Random Forest separately on real and synthetic data sets for subsequent performance comparison based on Accuracy, Precision, Recall and F1-Score metrics.

The evaluation of fidelity alongside functionality in the project establishes the capability of synthetic data to replace real data during analytics and artificial intelligence-driven operations. The framework demonstrates value as a deployable system because it manages scalability and explainability in addition to providing flexible operations in multiple domains.

6.2 Proposed Architecture

The devised algorithm creates domain-invariant synthetic data records that retain performance in subsequent machine learning systems. Placing realworld data into the process first entails processing datasets characterized by missing or incorrect information. The datasets undergo acute preprocessing before analysis through methods which impute missing values by mean or median or most prevalent value and encode categorical variables through onehot or label transformations depending on model needs. Feature space norming techniques normalize numerical variables while standardized techniques normalize the variables to establish consistent features and all outliers receive structured processing to counteract model bias (Fig. 2).

A data generator receives training from the preprocessed data. CTGAN (Conditional Tabular GAN) joins Gaussian Copula Models as the fundamental methods for data synthetic generation. The conditional tabular GAN known as CTGAN has specifically been designed for tabular data to model intricate data distributions alongside their interdependent relationships. Gaussian Copula allows a statistical model to analyze inter-variable correlations through multivariate Gaussian transformations combined with copula analysis.

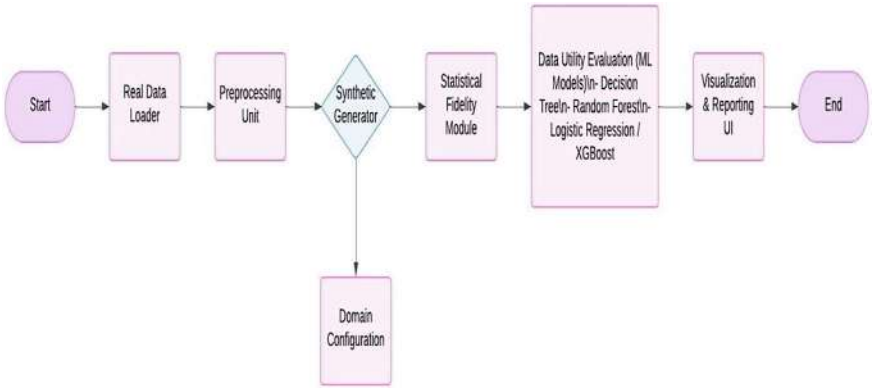


Fig. 2 Proposed architecture

The assessment process for synthetic data consists of two evaluation stages. The first evaluation tests statistical fidelity by performing distribution matching and correlation assessment as well as divergence metric analysis between actual and simulated data. The machine learning evaluation utilizes separate Decision Tree, Logistic Regression, Random Forest, and XGBoost modeling on both original and created synthetic data. The assessment of model performance involves comparison between accuracy, precision, recall and F1-score metrics that evaluate what degree synthetic data training models match the performance of real data counterparts. The workflow establishes synthetic data to reproduce real data structure together with statistics while maintaining prediction capabilities during actual application use.

Formulas Used

1. Jensen-Shannon Divergence (JSD):

$$JSD(P||Q) = \frac{1}{2}KL(P||M) + \frac{1}{2}KL(Q||M)$$

   where $M = \frac{1}{2}(P + Q)$
2. CTGAN Objective Function:

$$\min_G \max_D \mathbb{E}_{x\sim P_{real}}\left[\log D(x)\right] + \mathbb{E}_{z\sim P_z}\left[\log(1 - D(G(z)))\right]$$

3. Gaussian Copula Transformation:

$$\Phi^{-1}(F_i(x_i))$$

Preprocessing Techniques

a. The encoding method for categorical features involves one-hot or label encoding methods.

b. The normalization method used to standardize numerical features is either Min–Max scaling or StandardScaler.
c. The algorithm replaces missing numerical data points with median or mean values and replaces categorical items with the most common value.
d. Z-score or IQR-based filtering serves as the optional step for outlier handling.

Train-Test Split Strategy

a. The datasets present both real data and synthetic data divided into 30% testing data and 70% training data.
b. Training of ML models occurs through use of the training set while testing happens through evaluation on the testing set.
c. The model architecture is used identically with real data along with synthetic data to allow for equal utility evaluation.

7 Synthetic Data Generation

The synthetic data generation process in this project leverages two powerful techniques: CTGAN (Conditional Tabular GAN) and Gaussian Copulas.

CTGAN represents a Generative Adversarial Network that has been developed to address problems found in tabular data including different data types and distribution imbalance of categorical variables. CTGAN learns all conditional statistical relations between every variable compared to the others which results in improved synthetic data quality.

The Gaussian Copulas method transforms variables into multivariate Gaussian space through use of copulas that depict the dependency structure between variables. The procedure generates synthetic records which reproduce the existing correlations and maintain original statistical data features [7].

The first step trains models using preprocessed real data before the modeling process begins. The learned patterns and relationships guide the models to create synthetic data through the sampling process of their trained distributions. The synthetic datasets become available for assessment by statistical and ML-based assessment methods.

CTGAN (Conditional Tabular GAN)

- Handles both categorical and numerical data.
- Captures complex dependencies via conditional distributions.
- Robust for imbalanced datasets.

Gaussian Copulas

- The model implements copula functions to detect correlations between data features.
- The model changes features into Gaussian distribution space which makes sampling operations simpler.
- Ensures preservation of statistical structure.

Model Training

- The application of CTGAN models and Gaussian Copula models occurs during preprocessed real data training.
- Learn feature distributions and dependencies.

Synthetic Data Sampling

- Draw new synthetic data samples by using the trained models.
- Brand new data can be extracted from the simulation that matches the statistics of the original data.
- The procedure serves two purposes: utility evaluation as well as model testing.

8 Statistical Evaluation

Our evaluation of synthetic data statistical correctness combines both statistical tests with visual analysis approaches. The Kolmogorov–Smirnov (KS) test serves to detect statistical differences between real data distributions and synthetic data distributions of numerical features through distribution comparison analysis [8]. Jensen-Shannon Divergence (JSD) detects similarities between probability distributions of related features through its measurement process which provides advanced and symmetric assessment beyond traditional divergence methods. Distribution data plots created as histograms along with KDEs display the synthetic data performance relative to the real data characteristics.

Kolmogorov–Smirnov (KS) Test

- Compares cumulative distributions of real versus synthetic features.
- Identifies statistically significant differences.

Jensen-Shannon Divergence (JSD)

- Measures similarity between feature distributions.
- Symmetric distributions with bounded boundaries along with interoperable scoring characteristics.

Visual Distribution Comparison

- Histograms and Kernel Density Estimates (KDEs).
- A visual check should verify that all feature distributions match each other properly.

9 Results

Performance metrics of Decision Tree along with Random Forest and Logistic Regression and XGBoost models are presented in the above table for real data and synthetic data sets. All models show equivalent performance in the accuracy and precision as well as recall and F1 Score metrics which produces results of 0.70–0.71. The identical outcomes between real dataset models and synthetic dataset models demonstrates that synthetic data replication matches actual data in predicting abilities thus showcasing effective synthetic data generation practices. Models trained using synthetic data prove equally effective compared to those using real data according to this validation (Fig. 3 and Table 1).

The provided figures illustrate the statistical distributions of key features in a healthcare dataset, which are essential for evaluating the quality and fidelity of synthetic data. Figure 4 displays the distribution of blood pressure values, showing a unimodal shape centered around the average range, which is typical in real-world medical datasets. This indicates that the feature captures natural variance seen in patient populations. Figure 5 depicts the age distribution, revealing a slightly right-skewed pattern with most patients falling between the ages of 30–60, reflecting realistic demographic trends often observed in healthcare studies.

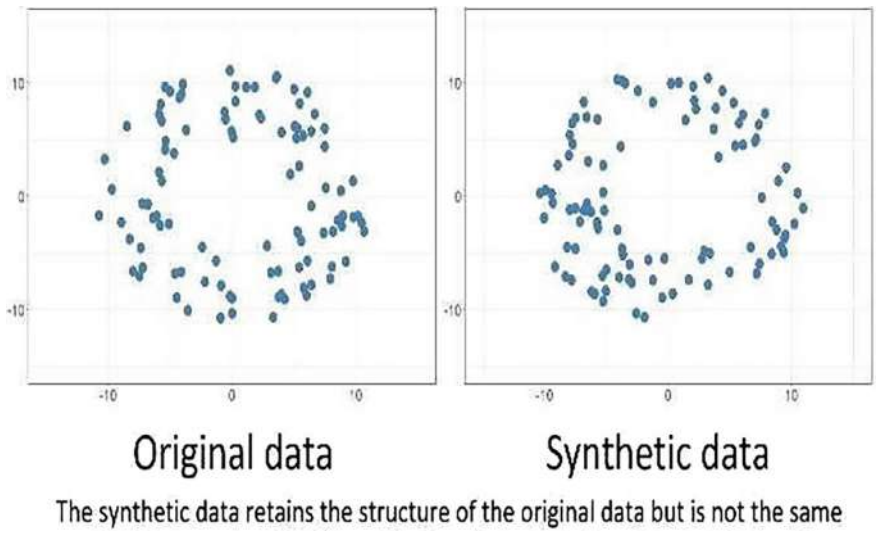


Fig. 3 A classification report based on the real versus synthetic dataset

Table 1 Performance metrics of Decision Tree, Random Forest, Logistic Regression, and XGBoost models

Model	Data type	Accuracy	Precision	Recall	F1 score
Decision Tree	Real	0.70	0.71	0.71	0.70
Random Forest	Real	0.70	0.71	0.71	0.70
Logistic Regression	Real	0.70	0.71	0.71	0.70
XGBoost	Real	0.70	0.71	0.71	0.70
Decision Tree	Synthetic	0.70	0.71	0.71	0.70
Random Forest	Synthetic	0.70	0.71	0.71	0.70
Logistic Regression	Synthetic	0.70	0.71	0.71	0.70
XGBoost	Synthetic	0.70	0.71	0.71	0.70

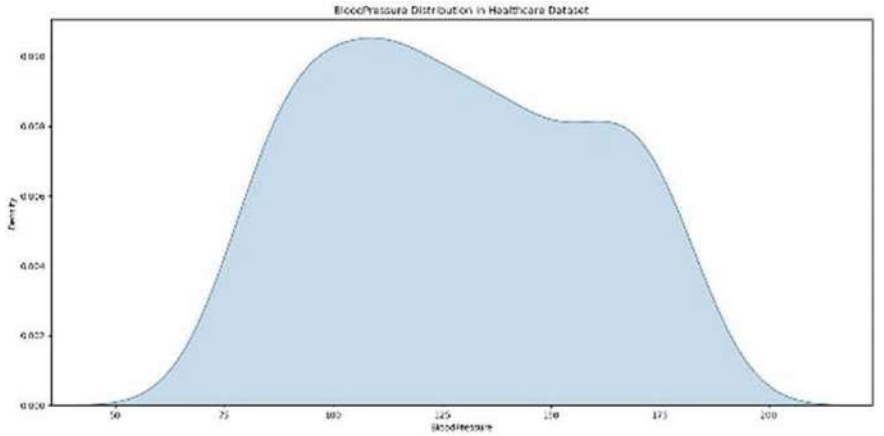


Fig. 4 Blood pressure distribution in synthetic healthcare dataset

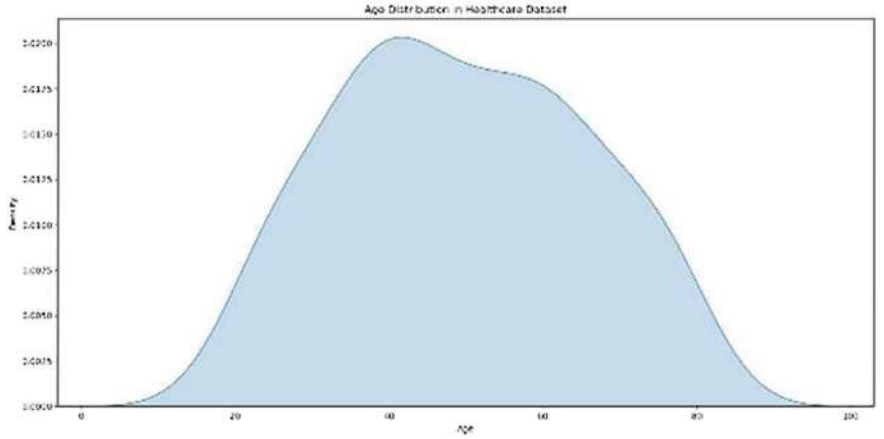


Fig. 5 Age distribution in synthetic healthcare dataset

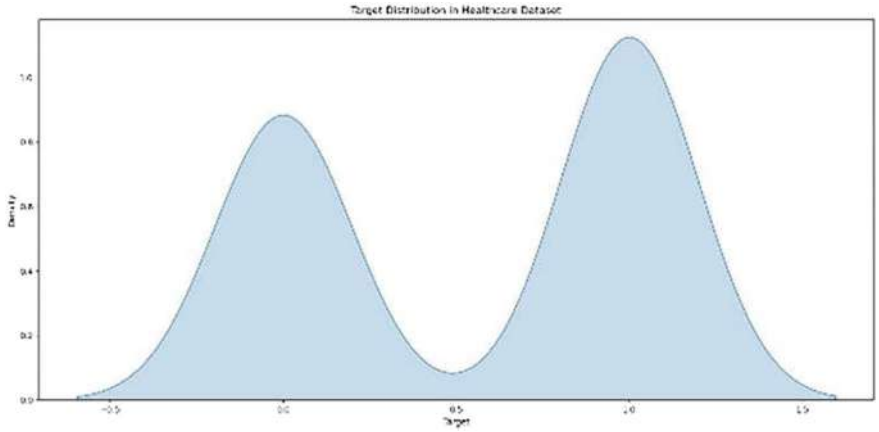


Fig. 6 Target variable distribution in synthetic healthcare dataset

Figure 6 shows the distribution of the target variable, likely representing a binary classification problem such as the presence or absence of a disease. The bimodal pattern confirms that the dataset maintains a clear distinction between classes, crucial for model training and evaluation. These visualizations are pivotal in assessing whether synthetic data generation methods, such as CTGAN or Gaussian Copulas, can replicate real-world statistical patterns accurately.

10 Use Case: Healthcare Domain

The generation of synthetic healthcare data maintains vital medical attributes that include age together with blood pressure measurements alongside the prediction target disease classes. The process begins with pre-processing real patient data before CTGAN and Gaussian Copulas models generate synthetic versions of high-fidelity quality. A distribution analysis of important features happens through visual inspection combined with statistical tests for ensuring realistic results. The evaluation process determines the synthetic data's practicality by developing Decision Trees, Random Forest, and XGBoost machine learning models on authentic as well as synthetic datasets. After training multiple choice prediction models the generated synthetic data undergoes an analysis of its predictive capabilities through comparison of model performance metrics including Accuracy and Precision along with Recall and F1 Score [9] (Table 2).

Table 2 Healthcare domain synthetic dataset generation

Age	Blood pressure	Cholesterol level	Diabetes risk
60	173	Normal	No
34	179	Low	Yes
44	162	High	No
23	157	High	Yes
83	97	Normal	Yes
77	150	High	Yes
53	134	Low	Yes
41	96	Normal	No
20	80	Low	Yes
35	124	High	Yes
43	139	Normal	No
25	164	Low	No
66	130	Normal	No
53	81	Normal	No
87	164	High	No
80	151	High	Yes
51	129	Low	No
84	119	Normal	Yes
69	177	High	No
45	125	High	No
41	139	High	Yes
26	81	High	No
73	178	High	Yes
81	98	Normal	No
61	105	Normal	Yes
75	179	Normal	No
75	101	Low	No
89	94	High	Yes
28	118	High	No
37	126	Low	No
54	82	High	Yes
52	174	High	No
32	87	Normal	Yes

11 Conclusion

This study combines generative models with a statistically supported and machine learning-driven assessment framework to provide a novel and flexible solution for the creation and evaluation of synthetic data. The main contribution is showing that models trained on real data can retain their predictive power and attain nearly identical statistical properties when using synthetic data. The system provides scalability and reusability across various domains thanks to its modular architecture. The evaluation results validated the synthetic data's real-world applicability by confirming that, after processing and modeling, it produced comparable performance across several classifiers. Furthermore, the pipeline promotes transparency, interpretability, and useful deployment by offering visualization components and domain-aware configurations. This work essentially demonstrates that synthetic data can be a trustworthy stand-in for real data when it is created and verified methodically, promoting privacy-preserving.

12 Future Scope

The present framework deals only with tabular data structures but upcoming improvements will enable the framework to process time-series data together with unstructured text and image information. Time-series data exists extensively in finance industries as well as healthcare monitoring and IoT networks so advanced models such as Time GAN or recurrent neural networks should be used to address their temporal dependencies. As an expansion to structured tabular data generation, the system becomes applicable to various real-world applications by implementing generative models for unstructured data such as GPT for text and GANs for images.

Future research needs to combine differential privacy tools into its framework. This addition will establish sophisticated privacy protection mechanisms that secure the synthetic output generation process by ensuring original data points cannot be traced from the final results. The implementation of this step establishes a connection between the framework and GDPR and HIPAA standards to enhance synthetic data trustworthiness in regulated settings.

References

1. Kossen, J., Nebgen, B., Willke, T.L.: Active learning for synthetic data generation with generative adversarial networks (2021). arXiv preprint arXiv:2103.10391
2. Choi, E., Biswal, S., Malin, B., Duke, J., Stewart, W.F., Sun, J.: Generating multi-label discrete electronic health records using generative adversarial networks. In: Machine Learning for Healthcare Conference (2017). arXiv preprint arXiv:1703.06490
3. Xu, L., Skoularidou, M., Cuesta-Infante, A., Veeramachaneni, K.: Modeling tabular data using conditional GAN. In: Advances in Neural Information Processing Systems, vol. 32 (2019)

4. Patki, N., Wedge, R., Veeramachaneni, K.: The synthetic data vault. In: IEEE International Conference on Data Science and Advanced Analytics (DSAA), pp. 399–410 (2016)
5. Yoon, J., Jarrett, D., van der Schaar, M.: Time-series Generative Adversarial Networks. In: NeurIPS, vol. 33 (2020)
6. Jordon, J., Yoon, J., van der Schaar, M.: PATE-GAN: generating synthetic data with differential privacy guarantees. In: International Conference on Learning Representations (ICLR) (2018)
7. Park, N., Ghosh, J.: Data synthesis based on generative adversarial networks: a survey. ACM Comput. Surv. **54**(3), 1–38 (2021)
8. Templ, M.: Artificial Data for Official Statistics: Synthetic Data Generation and Disclosure Control. Springer Nature (2023)
9. Goncalves, A., Ray, P., Soper, B., Stevens, J., Coyle, L., Sales, A.P.: Generation and evaluation of synthetic patient data. BMC Med. Res. Methodol. **20**(1), 108 (2020)

Safe-Voice-UPI for Secured Digital Transaction

Bhagwan Thorat, Mohini Pawar, Pranali Wankhade, Tanishka Patil, Sujal Pawar, and Vivek Patil

Abstract Ensuring transaction security is essential in the rapidly shifting world of digital payments. A deep learning-based voice authentication system called Safe-Voice-UPI attempts to fix the flaws in the Unified Payments Interface (UPI) framework of India, such as fraud, phishing, and illegal access. The project incorporates biometric voice recognition into the UPI system by utilizing distinctive vocal traits like pitch and tone, offering a reliable and convenient substitute for more conventional security measures like PINs. By providing improved fraud prevention, real-time monitoring, and user authentication, Safe-Voice-UPI has the potential to completely transform digital transaction security. This paper describes the development, integration, and testing methodologies of this system.

Keywords Voice authentication · UPI security · Classical linear algebra for digital payments

B. Thorat · M. Pawar (✉) · P. Wankhade · T. Patil · S. Pawar · V. Patil
Vishwakarma Institute of Technology, Pune, India
e-mail: mohini.pawar22@vit.edu

B. Thorat
e-mail: bhagwan.thorat@vit.edu

P. Wankhade
e-mail: pranali.wankhade22@vit.edu

T. Patil
e-mail: tanishka.patil22@vit.edu

S. Pawar
e-mail: sujal.pawar22@vit.edu

V. Patil
e-mail: vivek.patil22@vit.edu

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_12

1 Introduction

In India, digital transactions have been revolutionized by the Unified Payments Interface (UPI), which offers a quick, easy, and convenient payment experience. But as its use grows, UPI has come under attack from security risks like phishing scams, identity theft, and illegal access. There is an urgent need for more sophisticated and trustworthy authentication techniques because traditional security measures like PINs and passwords have built-in flaws. To overcome these obstacles, Safe-Voice-UPI presents a deep learning-based voice authentication system. The system generates customized voice profiles by utilizing each person's distinct voice qualities, which increases transaction security and enhances user convenience. In contrast to traditional techniques, voice-based authentication offers a biometric method for safe UPI transactions, allowing users to confirm payments using straightforward voice commands. By strengthening identity verification and lowering fraud, this integration seeks to establish a standard for safe digital payment ecosystems.

The process begins with collecting a diverse dataset of user voice samples, capturing unique vocal characteristics like pitch, tone, and speech patterns. Samples are recorded under varying conditions, including different accents, noise levels, and speech styles, ensuring adaptability. The data is pre-processed to remove noise and extract key features, such as Mel-Frequency Cepstral Coefficients (MFCCs), which are critical for distinguishing individual voices. These models are trained on the processed dataset to ensure high accuracy and resilience against challenges like background noise, voice mimicry, and tone variations.

Once developed, the voice authentication system is integrated into the UPI framework, replacing or augmenting traditional security measures like PINs. Users can approve transactions securely and conveniently through voice commands. The system undergoes extensive testing in real-world conditions to evaluate its performance, focusing on metrics such as accuracy, false acceptance rate (FAR), and false rejection rate (FRR). Robustness is tested against spoofing attempts and varying environmental conditions. Additionally, real-time transaction monitoring is implemented to detect suspicious activities and trigger instant alerts, ensuring enhanced security and fraud prevention. This methodology guarantees a reliable, efficient, and user-friendly voice authentication system for digital transactions.

2 Literature Survey

The paper [1] suggests a viable extension to the UPI design through the use of speaker recognition in a voice-model based on the FFT and 1-D CNN. The model reached an impressive accuracy rate of 98.46% on training data and 98% on validation data as it utilizes reliable acoustic features for authentication of users. The research also justifies its robustness by contrasting its performance with that of other methods like Mel Spectrogram and MFCC methods. Furthermore, the deployment of this model

into the UPI ecosystem is made possible through a structured interface, API and multi layered security. The paper highlights urgent issues in security and provides a good baseline for subsequent studies aimed at improving UPI security.

The paper [2] gives a thorough discussion on cyber risks associated with UPI and develops a multi-dimensional security strategy to meet these risks. The study addresses advancements in technology, such as using blockchain as an irrefutable record of trial and real time impulse detection, AI and machine learning for the internet of users, and's biometric verification for the end users. It draws from the literature, case studies, and expert consultations to provide practical measures for the improvement of UPI security. This research has great value in improving cyber security aspects of UPI system thereby enhancing the safety and reliability of the electronic payment system while enabling further development of digital financial transactions.

The paper [3] offers efficient contribution concerning the use and potential acceptance of UPI amongst Indian consumers through the application of the Diffusion of Innovation (DOI) theory. Recognized the factors of relative advantage, complexity, and observability as the foremost determinants that significantly work positively towards the intention of users to adopt UPI. Also, it shows that there's a good correlation between variables satisfaction, intention to use and recommend a product/service Statement. Instead of just looking at the operational factors, the research provides a holistic model to examine the dynamics of UPI adoption and also puts forward practical recommendations to increase user uptake and encourage wider usage of UPI.

This paper [4] study examines the concept of combining UPI with voice-driven interfaces ushering in a new era of "conversational commerce" in the digital payment scenario in India. It pinpoints the benefits of voice identification distribution, such as increased security, lower chances of phishing attacks, and easily use for people with sight problems and lower technological literacy and which promotes financial inclusion. The research importantly highlights the voice-based UPIs phenomenal adoption potential by speaking to user's feelings, use barriers, and the potential of local languages. This research highlights the tremendous possibilities of the interaction between voice technology and digital payments in their future development.

The paper [5] explains how UPI has changed the landscape for digital payments through seamless transactions using mobile numbers, QR codes & in-app chat. The whitepaper delves into the sophistication of UPI features especially, encryption and multi-factor authentication to maintain a high level of security and privacy for users. The study demonstrates the capacity of UPI in transforming digital payment systems through its usability and security, reiterating UPI's importance as a candidate for the future of digital economy.

3 Table of Analysis

Table of analysis

Paper	Key focus	Security features	Technologies used	Findings	Contribution to UPI	Limitations/ recommendations
[1] Enhancing UPI security using deep learning based voice authentication systems	Voice authentication for UPI security	98% accuracy in voice-based authentication using FFT and 1-D CNN	Deep learning (1-D CNN), FFT	High accuracy (98.46% on training, 98% on validation); compares with Mel Spectrogram & MFCC	Introduces robust voice authentication, improves UPI security	Further studies needed for real-world deployment in UPI ecosystem
[2] Cyber risks in UPI and multi-dimensional security strategy	Cyber risks and security measures for UPI	Blockchain, AI, machine learning, biometric verification	Blockchain, AI, ML, biometric verification	Proposes multi-dimensional strategy to enhance security, emphasizes AI and blockchain	Strengthens cyber security in UPI, improves transaction safety	Further practical implementations needed for blockchain integration
[3] Diffusion of innovation theory in UPI adoption	Factors affecting UPI adoption	N/A	DOI theory, user experience	Identifies relative advantage, complexity, and observability as key adoption factors	Helps improve UPI adoption, understanding user behavior	Focus on user-centric design for broader adoption
[4] Voice-driven UPI for conversational commerce	Voice-driven interfaces for UPI	Increased security, reduced phishing attacks, accessibility features	Voice identification, conversational commerce	Highlights the potential of voice-based UPI for wider adoption and financial inclusion	Enhances UPI's accessibility and security through voice	Further research on multi-language support and voice accuracy
[5] UPI's role in digital payments and security	UPI's role in digital payment systems	Encryption, multi-factor authentication	UPI, mobile numbers, QR codes, in-app chat	Demonstrates UPI's potential in transforming digital payments with advanced security features	Establishes UPI as a key player in the future of digital economy	Emphasizes the need for continuous innovation in security

4 Gap Analysis

4.1 Identified Gaps in Existing Research

The existing research on UPI security and voice-driven payment systems reveals several significant gaps that need to be addressed. While Doshi et al. explore FFT and 1-D CNN models for speaker identification, their application is primarily limited to generic layered security. The potential for integrating these models into real-world UPI systems for secure voice-based transaction authentication remains unexplored comprehensively [1].

Moreover, current works such as those by Gunti et al. and Ambadkar et al. focus on specific aspects of UPI security or usability but fail to propose an end-to-end system that integrates voice authentication, AI-driven fraud detection, and advanced security protocols into a cohesive framework [2, 5].

Additionally, studies like Mohan Kumar et al. focus on user attitudes toward voice-enabled payment systems but lack a technical or implementation-centric approach. Practical challenges, such as mitigating spoofing attacks, handling background noise, and ensuring consistent performance across diverse real-world scenarios, are insufficiently addressed [4].

Another critical gap is the absence of accessibility-focused studies. Although voice authentication is acknowledged for its potential to improve financial inclusion for users with low digital literacy or disabilities, little research investigates how such systems would impact adoption and usability for these demographics [4].

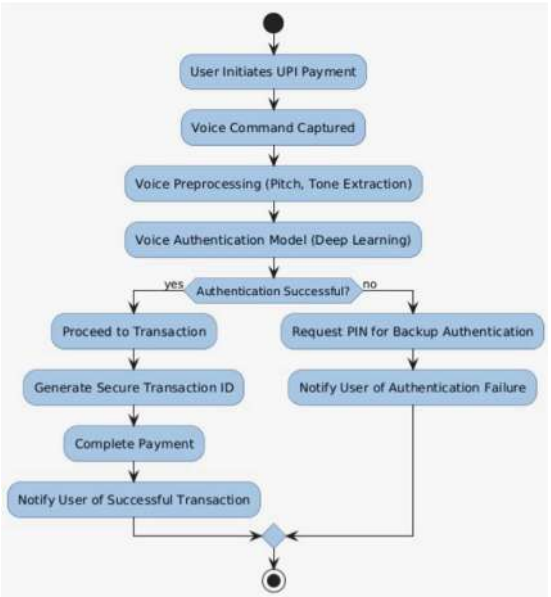
Finally, while Fahad et al. examine factors influencing UPI adoption broadly, there is limited analysis of the specific drivers and challenges associated with adopting voice-based systems, such as usability, security perceptions, and reliability [3].

4.2 Addressing the Gaps in This Research

This research aims to address the identified gaps by developing a deep learning-based voice authentication system that leverages FFT and 1-D CNN models specifically tailored for secure UPI transactions. These models will cater to the unique requirements of speaker identification in financial contexts, ensuring a robust and reliable system [6, 7]. To enhance real-world applicability, the system will be tested against practical challenges such as noise interference, spoofing attacks, and variations in voice due to factors like illness or emotions, ensuring it performs consistently in diverse scenarios. Additionally, this study proposes an end-to-end framework that integrates voice authentication with AI-driven fraud detection and layered security protocols, providing a seamless and secure transaction experience for UPI users [8].

Beyond technical advancements, this research emphasizes enhancing accessibility by exploring how voice-enabled UPI systems can improve financial inclusion for underrepresented demographics, such as users with low digital literacy or disabilities. The solution will be designed to be both user-friendly and inclusive. Finally, the study will analyze key adoption factors by applying user-centric research to understand the behavioural and technical elements that influence the acceptance of voice-based payment systems [9]. This holistic approach ensures the proposed solution aligns with user needs and addresses security, usability, and accessibility comprehensively.

5 Proposed System



Flowchart of safe-voice-UPI

6 Methodology

6.1 Data Description

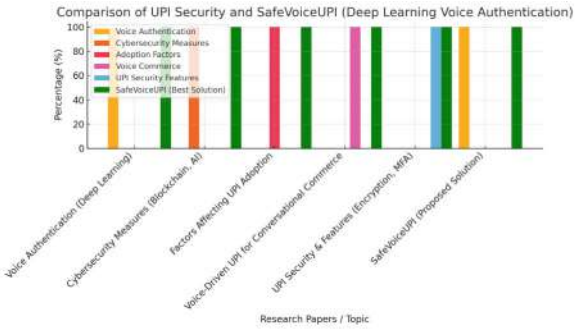
The process begins with collecting a diverse dataset of user voice samples, capturing unique vocal characteristics like pitch, tone, and speech patterns. Samples are recorded under varying conditions, including different accents, noise levels, and speech styles, ensuring adaptability. The data is pre-processed to remove noise and extract key features, such as Mel-Frequency Cepstral Coefficients (MFCCs), which are critical for distinguishing individual voices. Advanced deep learning models, including Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), are utilized to develop unique voice profiles. These models are trained on the processed dataset to ensure high accuracy and resilience against challenges like background noise, voice mimicry, and tone variations.

6.2 Preprocessing

Once developed, the voice authentication system is integrated into the UPI framework, replacing or augmenting traditional security measures like PINs. Users can approve transactions securely and conveniently through voice commands. The system undergoes extensive testing in real-world conditions to evaluate its performance, focusing on metrics such as accuracy, false acceptance rate (FAR), and false rejection rate (FRR). Robustness is tested against spoofing attempts and varying environmental conditions. Additionally, real-time transaction monitoring is implemented to detect suspicious activities and trigger instant alerts, ensuring enhanced security and fraud prevention. This methodology guarantees a reliable, efficient, and user-friendly voice authentication system for digital transactions.

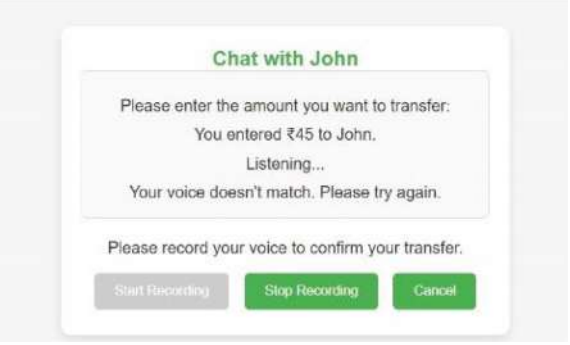
7 Result and Discussion

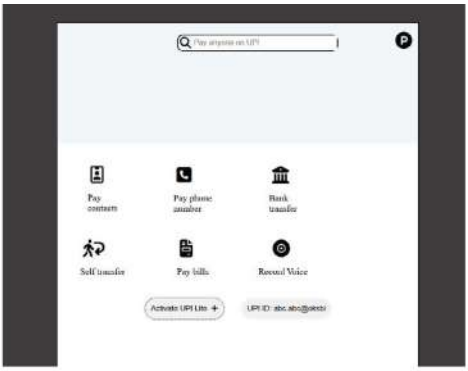
This table covers the major focus areas, findings, and contributions from each paper, along with their technologies, security features, and limitations or areas for improvement.

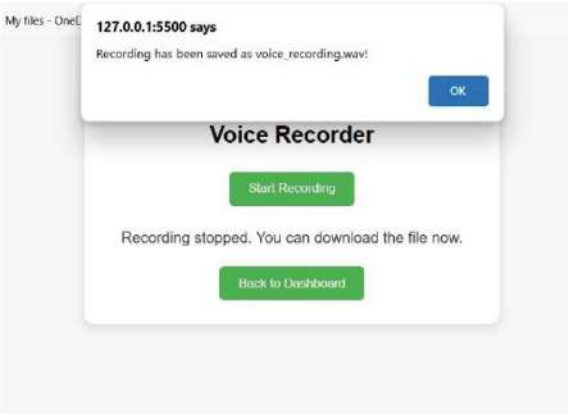


Comparison with other system

8 Sample Outputs







9 Scope of Research

The research topics of Unified Payments Interface (UPI) cover both technical, operational and user perspective to improve UPI effectiveness as a digital payment system. This includes harnessing cutting-edge security solutions for biometric authentication, blockchain integration, AI-driven fraud detection and risk mitigation of threats. Studies on user adoption and engagement can study factors that facilitate usage of online care such as perceived ease of use, accessibility issues, or low trust in the services (H2), while examining barriers to effective care including technological illiteracy or shortcomings in both access to mobile devices (such as smartphones) and internet infrastructure in rural areas. It helps usher in innovation by taking the lead in using new-age tech such as voice-enabled payments or adoption of regional language that can facilitate penetrating UPI further into hitherto unbanked sections of the population and enable financial inclusion Further, the analysis of UPI may include its economic impact and benefits with reference to assisting cashless economies as a whole.

10 Future Scope

The integration of deep learning-based voice authentication into the UPI ecosystem offers promising results, but there is substantial potential for future enhancements. The variability in user voices due to accents, languages, and environmental noise poses a significant challenge. Future research can focus on creating noise-robust models that maintain accuracy across diverse real-world conditions. Employing advanced techniques such as domain adaptation and transfer learning can enhance the system's ability to generalize across various demographic and linguistic groups. Furthermore, integrating additional features such as real-time liveness detection can address vulnerabilities to replay and synthetic voice attacks, further enhancing the security of digital financial transactions [10–14].

Multimodal authentication systems that combine voice with other biometrics, like facial recognition or fingerprint scanning, can be explored for a comprehensive security solution. Incorporating blockchain technology for securely storing and transmitting voice biometric data may improve data integrity and transparency. Another avenue for exploration involves leveraging quantum computing to develop advanced encryption techniques for secure authentication processes. Expanding this system to other financial platforms and global payment gateways can further solidify its impact, making it a universal standard for secure digital payments. Future work should also focus on real-world implementation and user acceptance studies to fine-tune the system's performance and usability.

11 Conclusion

This research presents a novel approach to addressing the critical security concerns of the UPI ecosystem through the implementation of a deep learning-based voice authentication system. With a training accuracy of 98.46% and a validation accuracy of 98%, the proposed model demonstrates high reliability in identifying users based on their voice biometrics. By leveraging CNN with Fast Fourier Transform for feature extraction, the system provides a secure, efficient, and user-friendly method for financial authentication. The integration of this model into the UPI framework not only strengthens security but also improves the overall user experience by enabling seamless and hands-free transactions.

References

1. Doshi, P.N., Khekare, G., Khetan, U.: Enhancing UPI security using deep learning based voice authentication systems. Int. J. Intell. Syst. Appl. Eng. (IJISAE) **12**(3), 2301–2311
2. Vijay, G., Reddi, D.S.K.: A study on enhancing the securities on UPI payments: exploring the measures and technology for secure transactions. Int. J. Res. Publ. Rev. **5**(6), 822–835 (2024)

3. Fahad, M.S.: A study on enhancing the securities on UPI payments: exploring the measures and technology for secure transactions. Digit. Bus. **2**, 100040 (2022)
4. Mohan Kumar, B.**,** Lakshmi, N., Kadakia, M.: A study on evolution of UPI towards a voice driven payment system—from taps to talks. J. Inf. Educ. Res. **4**(2). ISSN 1526-4726
5. Ambadkar, P., Ambure, S., Bhoir, H., Indalkar, S., Bhosale, V.: EZPAY—enhancing UPI usability and security. Int. Res. J. Mod. Eng. Technol. Sci. e-ISSN 2582-5208
6. Prachi, N.N., et al.: Deep learning based speaker recognition system with CNN and LSTM techniques. In: 2022 Interdisciplinary Research in Technology and Management (IRTM), pp. 1–6 (2022)
7. Khekare, G., Verma, P., Raut, S.: The smart accident predictor system using internet of things. In: Cloud IoT, pp. 163–175. Chapman and Hall/CRC (2022)
8. Singh, M.K.: A text independent speaker identification system using ANN, RNN, and CNN classification technique. Multimed. Tools Appl. Int. J., 1–13 (2023)
9. Barhoush, M., Hallawa, A., and Schmeink, A.: Robust automatic speaker identification system using shuffled MFCC features. In: 2021 IEEE International Conference on Machine Learning and Applied Network Technologies (ICMLANT), pp. 1–6 (2021)
10. Khekare, G., Gambhir, S., Abdulrahman, I.S., Kumar, C.M.S., Tripathi, V.: D2D network: implementation of blockchain based equitable cognitive resource sharing system. In: 2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, pp. 908–912 (2023). https://doi.org/10.1109/ICACITE57410.2023.10182834
11. Vassilev, V., et al.: Two-factor authentication for voice assistance in digital banking using public cloud services. In: Proceedings of the Confluence 2020—10th International Conference on Cloud Computing, Data Science and Engineering, pp. 404–409 (2020)
12. Madwanna, Y., Khadse, M., Chandavarkar, B.R.: Security issues of unified payments interface and challenges: case study. In: ICSCCC 2021—International Conference on Secure Cyber Computing and Communications, pp. 150–154 (2021)
13. Mohd Hanifa, R., Isa, K., Mohamad, S.: A review on speaker recognition: technology and challenges. Comput. Electr. Eng. (2021)
14. Khan, S.A., Naaz, S.: Comparative analysis of finger vein, iris and human body odor as biometric approach in cyber security system. In: 2020 2nd international conference on innovative mechanisms for industry applications (ICIMIA), pp. 525–530 (2020)

Phishing Site Analyzer: AI-Driven Real-Time Detection with MLP and Flask

Y. Kranthi Kumar, Harsh J. Shah, Kola Aravind, Pandipati Mokshagna, and Talluri Subrahmanyam

Abstract This research introduces an Advanced Phishing Detection System integrating feature extraction, EDA (Exploratory Data Analysis), and machine learning for real-time threat detection. It evaluates four AI models—Random Forest, SVM, XGBoost, and MLP—achieving high accuracy in identifying phishing websites. Traditional rule-based methods struggle with zero-day attacks, while ML models like Decision Trees and Logistic Regression fail to handle complex patterns effectively. Using a publicly available phishing dataset, the system extracts key features like URL length, domain age, and HTTPS presence. MLP and XGBoost achieve the highest accuracy (99.03% and 95.73% for MLP). A Flask-based web app enables real-time detection. Future work includes LSTMs for sequential URL analysis and continuous learning for adaptive phishing defense. The novelty of this work lies in combining deep learning with real-time deployment via Flask, using a diverse set of phishing indicators beyond basic URL features.

Keywords Feature extraction · Flask interface · Machine learning · Phishing detection · Exploratory data analysis (EDA) · Web security

Y. Kranthi Kumar · H. J. Shah (✉) · K. Aravind · P. Mokshagna · T. Subrahmanyam
Computer Science and Engineering (Artificial Intelligence and Machine Learning) Department, Lakireddy Bali Reddy College of Engineering, Mylavaram, Andhra Pradesh, India
e-mail: harshjshah04@gmail.com

Y. Kranthi Kumar
e-mail: kranthi@lbrce.ac.in

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_13

1 Introduction

1.1 Background

Phishing attacks, which target people and organizations to obtain sensitive information including login passwords, financial information and personal data have emerged as one of the most common cybersecurity dangers. Phishing websites are used by cybercriminals to trick consumers into submitting their credentials by resembling authentic ones. Due to their inability to identify novel and changing phishing attacks, traditional security measures like rule-based detection techniques and block lists have limitations. There is an urgent need for advanced automated devices that can identify phishing messages in real-time due to the increasing complexity of attacks.

Through the analysis of website attributes and the discovery of hidden patterns suggesting of fraudulent activity machine learning has become an effective instrument in phishing detection. To distinguish between real and fraudulent websites AI-based algorithms use a variety of characteristics including URL structure, domain age, HTTPS presence and text analysis. Despite improvements current models frequently have significant false-positive rates and are not flexible enough to handle novel phishing strategies. Thus, creating an effective phishing detection system that integrates several AI models and exploratory data analysis (EDA) can improve the accuracy and reliability of detecting phishing attacks.

1.2 Research Motivation

Phishing attacks have emerged as a significant cybersecurity threat due to the quick growth of online services resulting in identity theft, data breaches and financial losses all over the world. New phishing websites can be created dynamically to avoid detection, making traditional detection techniques like rule-based systems and blacklists ineffective against them. The increasing complexity of phishing tactics such as social engineering and obfuscation need a more clever and flexible approach to detection. Investigating cutting-edge AI-driven methods for precise and real-time phishing detection is essential since using machine learning to assess and identify phishing patterns can greatly improve security measures.

1.3 Limitations of Prior Works

Prior phishing detection models frequently depend on antiquated methods such as rule-based systems or basic algorithms for machine learning which are not very effective in addressing changing phishing strategies. These models may not be able to identify advanced or novel phishing websites that employ advanced masking methods

due to their high false-positive rates. Furthermore, a lot of current models are based on static datasets which limits their ability to adapt to the constantly developing nature of phishing attempts. Additionally, they frequently overlook complex and detailed signs of phishing in favour of concentrating on a small number of characteristics which reduces their overall efficiency and precision in practical situations.

- The user experience and trust may be damaged by existing systems high false-positive rates which incorrectly identify reliable websites as phishing sites.
- They are less able to adapt to new changing phishing techniques since they rely on old datasets or static rule-based methodologies.
- Many algorithms neglect the wider range of hidden signals that could more accurately detect phishing attempts rather than of concentrating just on a limited set of attributes.
- Traditional systems detection processes might be slow, making it impossible to evaluate phishing threats in real time or react quickly to them.

1.4 Objectives

- Creating and put into use an advanced phishing detection tool that combines machine learning models, exploratory data analysis (EDA), and feature extraction for increased accuracy.
- To assess the efficiency of many machine learning models in identifying phishing websites such as Random Forest, MLP, XGBoost and SVM.
- Identifying phishing patterns by extracting important properties from internet data such as domain information, content-based attributes and URL characteristics.
- To use Flask to deliver real-time phishing detection with an easy-to-use interface that enables prompt feedback and threat assessment.

1.5 Main Contribution

The primary contribution of this research is the development of an AI-driven phishing detection system that combines feature extraction, exploratory data analysis (EDA), and machine learning models to effectively identify phishing websites. Among the models evaluated—Random Forest, SVM, XGBoost, and MLP—the Multilayer Perceptron (MLP) demonstrated the highest accuracy in both training and testing phases. To ensure practical usability, the system is deployed through a lightweight Flask-based web application that enables real-time URL analysis. This integration of deep learning with real-time detection offers a robust, accurate, and user-friendly solution to combat evolving phishing threats, addressing the limitations of traditional rule-based and static detection methods.

1.6 Novelty of the Proposed Approach

The novelty of this research lies in the integration of a deep learning-based Multilayer Perceptron (MLP) model with a real-time web-based detection system, enhanced by comprehensive feature extraction and Exploratory Data Analysis (EDA). Unlike prior studies that either rely on static machine learning models or focus on a limited set of URL-based features, this work employs a more diverse feature set—including domain-based and content-based attributes—to improve detection accuracy. Additionally, while many previous models demonstrate strong offline performance, they lack real-time deployment capability. This gap is addressed through the implementation of a lightweight Flask application that delivers fast, accurate phishing detection via an intuitive web interface.

Key novel contributions include:

- Deployment of an MLP model with superior performance in recognizing complex phishing patterns, achieving 95.73% testing accuracy.
- Use of EDA for understanding feature correlations and patterns before model training, improving model reliability.
- Real-time phishing URL detection through a user-friendly Flask interface—bridging the gap between research and practical application.
- A balanced evaluation of four models under the same system, providing a fair comparative analysis.

These aspects collectively differentiate the proposed system from conventional approaches and contribute to its effectiveness in detecting sophisticated phishing attacks.

2 Literature Survey/Related Work

The majority of previous phishing detection research has gone toward creating and improving systems to detect phishing attempts using different approaches. Heuristic techniques and rule based systems which employed established rules to identify phishing based on recognized patterns and signatures [1] were the mainstays of early attempts. Shahrivari et al. proposed the internet has made it possible for hackers to trick victims through social engineering and spoof websites a practice known as phishing [2]. Because machine learning and these assaults have similar traits machine learning is an effective way to identify them. In order to anticipate phishing websites this research examines the outcomes of many machine learning techniques. Rashid et al. provides an effective machine learning based phishing detection method that uses just 22.5% of novel functionality to correctly identify 95.66% of phishing and suitable websites. When the method [3] is combined with a support vector machine classifier it performs well when tested against common phishing datasets from the

University of California Irvine collection [4]. Gandotra et al. examines feature selection techniques [5] for phishing website detection and finds that despite the time-consuming aspect of creating a large number of features random forest reduces model building time without sacrificing accuracy. Nimeh et al. using a data set of 2889 authentic and phishing emails this study examines machine learning methods [6] for phishing email prediction. 43 characteristics are used for classifier training and testing (Table 1).

Tang et al. conducted a comprehensive survey on ML-based phishing detection methods, highlighting the use of Random Forests and Support Vector Machines for URL-based classification. **Sahingoz et al.** proposed an NLP-based system that achieved 97.98% accuracy using seven classification algorithms but was limited by dataset diversity and lack of real-time deployment. **Alazaidah et al.** evaluated 24 classifiers and identified Random Forest and J-48 as top performers; however, their evaluation was restricted to only two datasets.

Our approach builds upon these works by:

- Combining **feature extraction and EDA** to better understand and visualize the phishing patterns.
- Evaluating **four different models** (Random Forest, SVM, XGBoost, and MLP) under the same framework for fair comparison.
- **Deploying the system in real-time** using Flask, whereas most prior work focused only on offline model performance [10].
- Demonstrating that **MLP outperforms** traditional ML methods in detecting complex, obfuscated phishing websites [11].

Table 1 Literature survey

Author	Techniques	Merits	Demerits
Tang et al. [7]	Machine learning-based phishing website detection, data collection, feature extraction, modeling, evaluation	Provides a comprehensive survey and comparison of various anti-phishing methods	Focuses mainly on phishing link detection, may not cover all phishing attack types comprehensively
Ozgur Koray Sahingoz et al. [8]	The system uses seven classification algorithms and NLP-based features for phishing detection	It offers language independence, real-time execution, and high accuracy (97.98%) in phishing detection	It may not address all types of phishing attacks and may be limited by the dataset used
Alazaidah et al. [9]	24 classifiers representing 6 learning strategies, 4 feature selection methods	Identified best classifiers (Random Forest, Filtered Classifier, J-48) and feature selection method (Info Gain Attribute Evaluation)	Limited to two datasets, might not generalize to other phishing attack types

Unlike previous systems, our method balances **accuracy, adaptability, and usability**—making it suitable for both research and practical deployment. The integration of a deep learning model (MLP) with a real-time web interface makes our solution more robust in dynamic cybersecurity environments.

3 Data Collection and Preprocessing

The quality of training and assessment data significantly impacts the efficiency of phishing detection systems. Data collection and preprocessing are crucial for ensuring model accuracy and resilience. This process involves gathering diverse instances from both malicious and legitimate sources using public phishing datasets, collaborative data-sharing platforms, and web scraping tools [12]. The dataset includes URLs classified as authentic or phishing, along with metadata like page layout, content attributes, and domain registration details. Feature extraction transforms unstructured data into a usable format by analyzing domain age, SSL certificate status, URL length, special characters, and HTML content attributes (e.g., login forms, iframes). The algorithm detects phishing traits using these features. Exploratory Data Analysis (EDA) helps identify patterns and potential biases using statistical summaries, correlation analysis, and visualizations [13]. Preprocessing includes data cleaning (handling missing values, removing duplicates, and resolving label inconsistencies) to ensure reliability. Normalization techniques like z-score scaling standardize feature magnitudes for better model performance [14].

4 Principles and Methods

The methods of phishing detection system form the basis of its reliability and accuracy. The system's primary objective of quickly recognizing and preventing phishing assaults is achieved using a variety of machine learning techniques and artificial intelligence models. This part outlines the key concepts and methods used in the development and implementation of the detection system. The aim of the phishing detection system are machine learning approaches that enable the models to gain insight from data and generate predictions. Systems must be trained for machine learning to find patterns in the dataset. These algorithms are able to recognize characteristics linked to phishing by analyzing past data (Fig. 1).

A strong the extraction and selection of features procedure is necessary for efficient phishing detection. Finding and measuring pertinent characteristics from unprocessed data such as URL length, special character presence, HTML content structures and domain information is known as feature extraction. In this stage unstructured data is converted into a format that can be processed by machine learning models. By selecting the most important characteristics that increase the prediction capacity of the model feature selection improves this procedure even further.

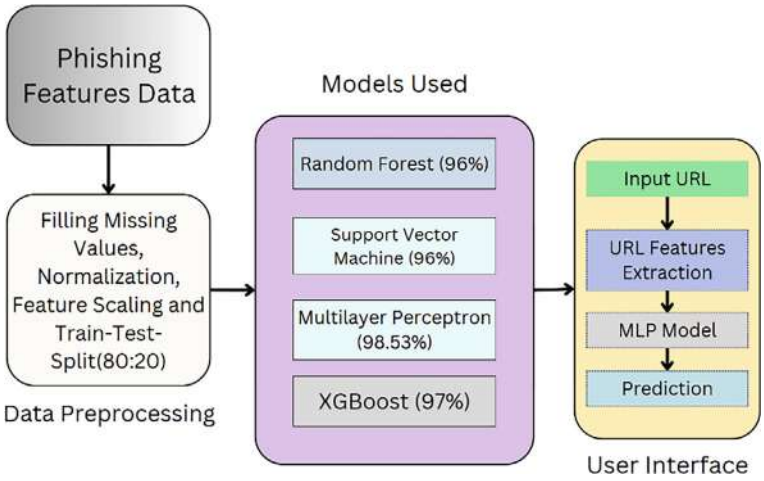


Fig. 1 Architecture of system

In order to make sure that the model focuses on the most important qualities techniques like feature priority ranking, correlation analysis and dimensionality reduction (e.g., Principal Component Analysis) are utilized which improves accuracy and efficiency. The system enhances detection capabilities by utilizing a variety of machine learning models. With different techniques to prediction and categorization, each model plays to its strengths. While Random Forests combine many decision trees to increase accuracy and resilience Decision Trees offer comprehensible decision rules based on feature values.

4.1 Justification of the Proposed Approach

The Multilayer Perceptron (MLP) was selected for its strong ability to capture complex phishing patterns that simpler models often miss. Combined with EDA and feature extraction, it improves detection accuracy. Flask was used to deploy the model in real-time, making the system practical, lightweight, and user-friendly. This approach balances performance, interpretability, and real-world usability.

4.1.1 Exploratory Data Analysis

Before using any machine learning models, the data analysis process must first do exploratory data analysis (EDA) which entails examining and comprehending the dataset. Using statistical and graphical techniques EDA aims to identify patterns, identify anomalies, test hypotheses and verify assumptions. This research makes

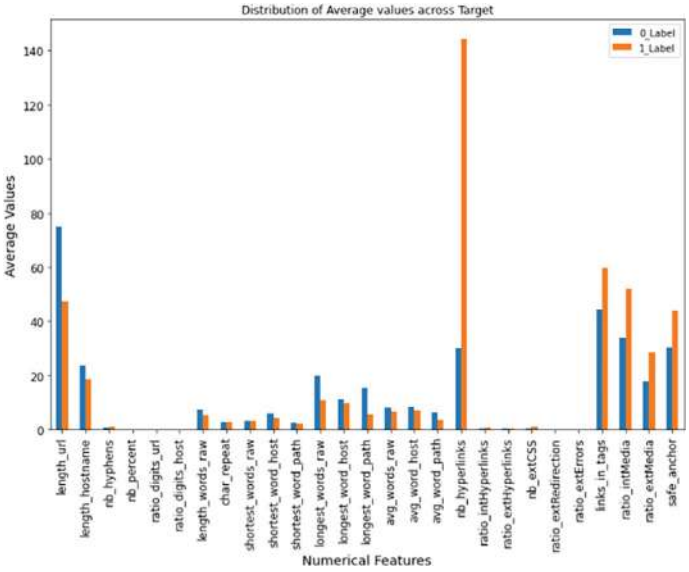


Fig. 2 Distribution of average value versus features

it possible to make accurate choices regarding choosing features, data preparation, and model building. This is an in-depth analysis of the EDA [15] process's main stages. The main stage in the EDA process is collecting and integrating data from several sources. For a phishing detection system this involves gathering data on URLs, website content and related information. Working with public databases online scraping, or data exchange services can all produce data. Integration is the process of combining data from multiple sources into one collection while preserving format and architectural integrity. This phase is essential because it provides an in-depth evaluation of the data and creates the foundation for future studies (Figs. 2 and 3).

Data visualization that includes turning the data into visual representations that demonstrate correlations, patterns and trends is an effective tool in EDA. A range of visualizations are used including as histograms to show the distribution of numerical values, box plots for recognizing outliers, scatter plots to examine feature relationships and bar charts for evaluating categorical data. Visualizations make it easier to understand big data and identify trends or abnormalities than statistics summaries. Correlation analysis [16] analyzes the connections between different features to identify main relationships or correlations. Techniques such as Pearson correlation coefficients are used to measure the strength and direction of linear relationships between numerical data. Tables of variables or the chi-square tests can be used to assess connections for categorical data. Finding identical components and choosing the most important ones for model construction are made simple by an understanding of relationships [17].

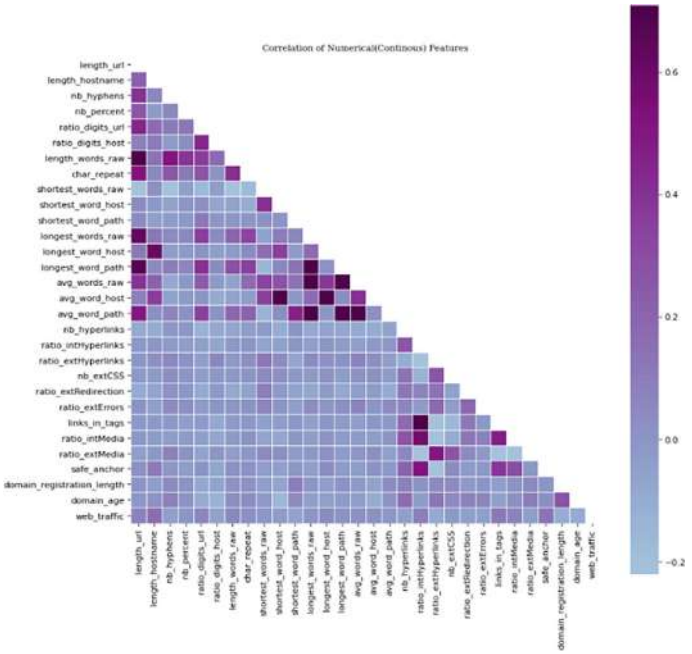


Fig. 3 Correlation of features

4.1.2 Machine Learning Models

A. ***Random Forest***

An ensemble learning system called Random Forest combines several decision trees to provide predictions that are more reliable and accurate. It is a member of the bagging algorithm family in which every tree is developed on a distinct subset of data that is selected at random using replacement. A major issue with each decision tree is overfitting which is lessened by this method called bootstrap aggregation. The final prediction in a random forest is determined by taking the majority of the votes for tasks such as classification or by average the predicted outcomes of all trees for tasks involving regression. Each tree in the random forest is constructed independently. By reducing the variation that particular decision trees may experience this ensemble technique helps smooth out estimates and improves generalization on unknown data.

$$\hat{y} = \text{mode}(\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n)$$

where $\hat{y}$ is the prediction and N is the number of decision Trees. The capacity of Random Forest to manage complex datasets and maintain high accuracy even in the missing values is one of its main features. Additionally, it offers information on the

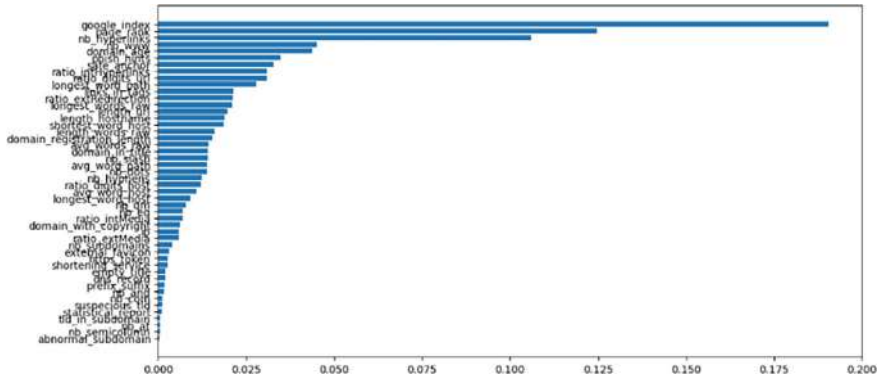


Fig. 4 Feature importance (Random Forest)

corresponding importance of each feature in the process of making choices which is helpful when choosing features for more complex models.

Random Forest employs a technique called feature bagging [18] in which a random collection of characteristics is chosen for analysis at each tree split. This enhances the resilience of the entire model by further decorrelating the component trees. Because numerous trees must be trained, Random Forest can be analytically costly yet it is nevertheless quite effective in a variety of applications including classification jobs. Every leaf node in the tree represents an outcome every branch in the tree represents a decision rule and every node in the tree represents a feature. In order to divide the data into discrete categories the model iteratively creates branches by dividing the data according to the characteristic that offers the best separation. The Random Forest of the phishing detection system obtained training accuracy of 96.58%. The Random Forest may experience overfitting if it is not appropriately pruned, while being simple to understand and visualize. The Random Forest algorithm is renowned for its resilience [19] and capacity to manage huge datasets with a high degree of dimension (Fig. 4).

B. *XG-Boost*

The gradient boosting approach which is frequently used for learning projects like regression and classification is powerfully and effectively implemented by XGBoost (Extreme Gradient Boosting) [20]. Gradient boosting is the process of creating a sequence of decision trees each of which aims to fix the mistakes of the one before it. Each new tree in the XGBoost algorithm is constructed sequentially with an emphasis on the residual errors the discrepancies between the true and forecasted values of the preceding trees. The model is trained by reducing a loss function that incorporates a regularization term that regulates the model's complexity and prevents overfitting as well as the error of the model's predictions. It uses a number of strategies to accelerate training including sparsity-aware algorithms that efficiently manage missing data and parallelization which enables it to employ several cores for computing.

$$F_m(x) = F_{m-1}(x) + \eta \cdots h_m(x)$$

$F_m(x)$ is the new model,
$F_{m-1}(x)$ is the previous model,
η (learning rate) controls step size,
$h_m(x)$ is the weak learner (decision tree).

The trees minimize a **loss function** plus a **regularization term**:

$$L = \sum_{(i)} I(y_i, \hat{y}_i) + \sum_{(k)} \Omega(T_k)$$

$l(y_i, \hat{y}_i)$ is the loss (e.g., squared error for regression),
$\Omega(T_k)$ penalizes model complexity.

In order to lower computational costs the approach also makes use of sophisticated tree building techniques including split finding and approximate tree learning. In order to avoid needless computation and overfitting XGBoost also enables early stopping, which stops training if the model's success rate on a validation set begins to deteriorate. The XG-Boost algorithm in the phishing detection system had the second highest accuracy, with 97.0% for training and 95.10% for testing. XG-Boost is a popular choice for a range of applications involving machine learning due to its excellent performance and flexibility.

C. ***Support Vector Machine***

One kind of supervised learning technique used for regression and classification problems is called Support Vector Machines (SVM) [21]. Although it may also be used for regression problems the Support Vector Machine (SVM) is a strong and adaptable supervised learning technique that is mainly employed for classification. Finding a hyperplane that optimally divides the data into discrete classes while optimizing the margin between each class's nearest points known as support vectors is the fundamental concept of support vector machines (SVM). The approach handles nonlinearly separable data by employing a mathematical technique known as the kernel trick to translate data points into higher-dimensional spaces. Without the requirement for explicit mapping SVM may function effectively in high-dimensional spaces thanks to the kernel function such as the polynomial kernel or radial basis function (RBF).

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i K(x_i, x) + b$$

b is the bias term,
y_i is the class label (+ 1 or − 1),
x_i are feature vectors. and $K(x_i, x)$ is the kernel function (e.g., RBF, polynomial).

SVM's capacity to handle high-dimensional information where numerous other approaches would fail is one of its main advantages. It works especially effectively for tasks like recognition of images, classification of text and biology that have defined margins of separation. Strong theoretical foundations of SVM such as the ideas of structural risk reduction aid in lowering overfitting and enhancing model adaptation. The SVM model in the phishing detection system obtained 96.54% training accuracy.

5 Proposed Methodology

5.1 Multilayer Perceptron

Multilayer Perceptron (MLP) was chosen as the proposed methodology due to its superior ability to capture complex patterns in phishing data, leading to higher accuracy compared to other machine learning models. The ability of MLP to approximate complex functions makes it particularly suitable for detecting phishing websites, which often employ obfuscation techniques to evade detection.

$$z^{(1)} = W^{(1)}a^{(1-1)} + b^{(1)}$$
$$a^{(1)} = f(z^{(1)})$$

$W^{(l)}$ = weight matrix for layer l
$b^{(l)}$ = bias vector for layer l
$z^{(l)}$ = pre-activation value
$a^{(l)}$, $a^{(l-1)}$ = activated output.

Multiple layers of nodes or neurons arranged into a layer of inputs one or more hidden layers and a layer of output make up a Multi-Layer Perceptron (MLP) [22] a type of forward artificial neural network. In the MLP every layer is completely coupled to the one behind it which means that each neuron in one layer has connections to every other layer's neuron. After being received by the input layer the data is transferred via any number of hidden layers where each neuron determines its output by applying an activation function and a weighted sum of the inputs. Until the data enters the end result layer where the final classification or prediction is formed this procedure is repeated throughout each hidden layer. A special type of network called Multilayer Perceptron has one or more hidden layers in between the input and output layers [23] (Fig. 5).

Their capacity for approximating complex functions is one of MLPs main advantages. An MLP may potentially learn to represent any continuous operation if it has enough hidden levels and neurons which makes it extremely adaptable for a variety of purposes. The number of layers the number of neurons in each layer the activation function and the learning rate are among the hyperparameters that must be adjusted

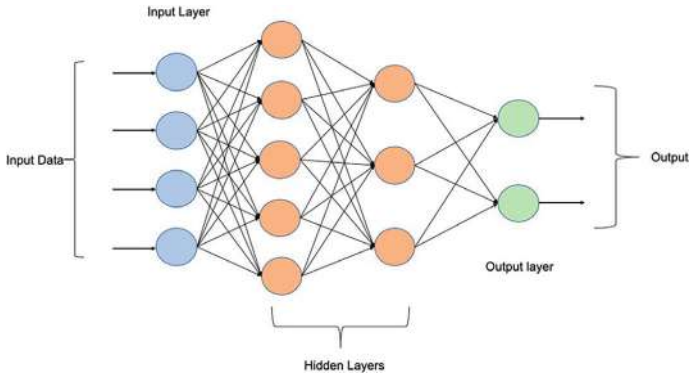


Fig. 5 MLP architecture

when training an MLP. MLPs are frequently employed in domains such as image recognition, language processing and economic prediction because they can handle both classification and regression issues. When it comes to finding irregular relationships MLPs are useful. The MLP model obtained training accuracy of 99.03% and testing accuracy of 95.73%.

6 Results

The phishing detection system was evaluated using Random Forest, SVM, XGBoost, and MLP. MLP outperformed the others, achieving 99.03% training accuracy and 95.73% testing accuracy, indicating strong generalization to unseen phishing instances. XGBoost also performed well (97.0% training, 95.10% testing accuracy), while Random Forest and SVM achieved over 96% accuracy, demonstrating their reliability.

MLP's superior performance stems from its deep learning architecture, which effectively captures intricate phishing patterns often missed by traditional models. It minimizes false positives and negatives, ensuring accurate detection while preventing unnecessary alerts.

For real-time usability, the system integrates a Flask-based interface, allowing users to analyze URLs within seconds. This accessibility makes it practical for both individuals and organizations to enhance cybersecurity defenses (Figs. 6, 7, 8, 9, 10 and 11; Table 2).

The Flask-based application enables real-time phishing detection, offering quick and accurate URL analysis for proactive security. Its speed, efficiency, and lightweight design make it ideal for cybersecurity applications. The intuitive interface ensures ease of use, promoting widespread adoption. By combining deep

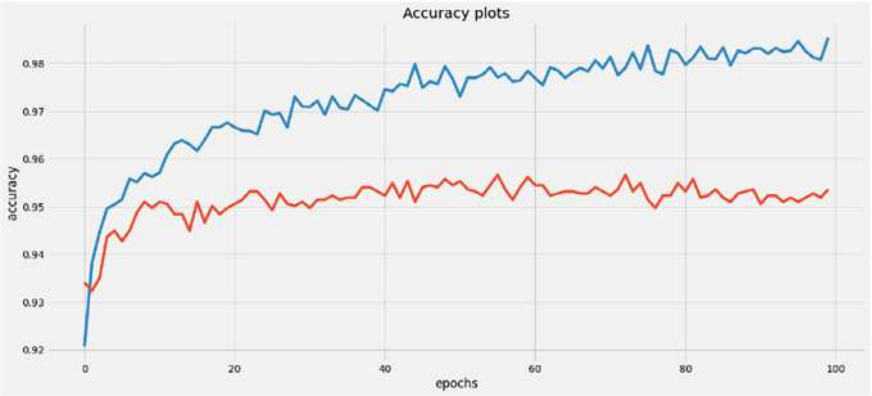


Fig. 6 Accuracy versus epochs of MLP

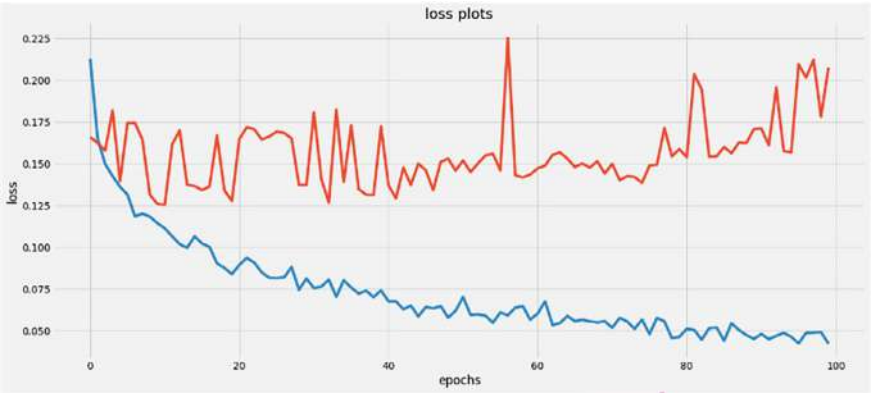


Fig. 7 Loss versus epochs of MLP

Fig. 8 Giving a phishing URL

Fig. 9 Real time detection (output)

Fig. 10 Giving a legitimate URL

learning with a responsive platform, the system enhances digital safety and reinforces AI-driven security solutions.

7 Conclusion

This study presents a comprehensive phishing detection system that not only achieves high accuracy using machine learning models—particularly Multilayer Perceptron (MLP)—but also ensures real-time usability through a lightweight Flask-based web

Fig. 11 Real time detection (output)

Table 2 Models accuracy on train and test data

Model	Train accuracy	Testing accuracy
Random Forest	96.58	94.18
MLP	99.03	95.73
XG-Boost	97.0	95.10
SVM	96.54	93.27

interface. By combining feature extraction, EDA, and deep learning techniques, the system effectively overcomes the limitations of traditional methods, offering a reliable and adaptive solution against evolving phishing threats. The integration of real-time detection and user accessibility makes this approach highly practical for enhancing cybersecurity at both individual and organizational levels. Future enhancements could integrate LSTM networks for sequential URL analysis and continuous learning for adapting to evolving threats. This study highlights AI's role in strengthening digital security and the importance of automated phishing detection in mitigating cyber risks.

References

1. Moghimi, M., Varjani, A.Y.: New rule-based phishing detection method. Exp. Syst. Appl. **53**, 231–242 (2016)
2. Shahrivari, V., Darabi, M.M., Izadim, M.: Phishing detection using machine learning techniques. arXiv preprint arXiv:2009.11116 (2020)
3. Rashid, J., et al.: Phishing detection using machine learning technique. In: 2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH). IEEE (2020)

4. Goldberg, S.M., McAdam, T.: A collaborative approach for processing electronic resources at the University of California, Irvine. Tech. Serv. Q. **20**(2), 21–32 (2002)
5. Gandotra, E., Gupta, D.: An efficient approach for phishing detection using machine learning. Multimed. Secur. Algorithm Dev. Anal. Appl., 239–253 (2021)
6. Abu-Nimeh, S., et al.: A comparison of machine learning techniques for phishing detection. In: Proceedings of the Anti-phishing Working Groups 2nd Annual eCrime Researchers Summit (2007)
7. Tang, L., Mahmoud, Q.H.: A survey of machine learning-based solutions for phishing website detection. Mach. Learn. Knowl. Extraction **3**(3), 672–694 (2021)
8. Sahingoz, O.K., et al.: Machine learning based phishing detection from URLs. Exp. Syst. Appl. **117**, 345–357 (2019)
9. Alazaidah, R., et al.: Website phishing detection using machine learning techniques. J. Stat. Appl. Probab. **13**(1), 119–129 (2024)
10. Lim, H., Sim, D., Choo, J.: EXPLICATE: phishing detection with explainable models and LLM-powered interpretability. arXiv preprint arXiv:2404.06393 (2024); EDA, OF: Exploratory data analysis. In: Handbook of Psychology, Research Methods in Psychology, vol. 2, p. 34 (2012)
11. Kulkarni, A., Zeng, H., Chang, M.-W.: PhishOracle: robustness evaluation of phishing webpage detection models via adversarial example generation. arXiv preprint arXiv:2403.07091 (2024)
12. Chatfield, C.: Exploratory data analysis. Eur. J. Oper. Res. **23**(1), 5–13 (1986)
13. Massaro, D.W., Friedman, D.: Models of integration given multiple sources of information. Psychol. Rev. **97**(2), 225 (1990)
14. Lowry, P.B., Gaskin, J.: Partial least squares (PLS) structural equation modeling (SEM) for building and testing behavioral causal theory: When to choose it and how to use it. IEEE Trans. Prof. Commun. **57**(2), 123–146 (2014)
15. Denisko, D., Hoffman, M.M.: Classification and interaction in random forests. Proc. Natl. Acad. Sci. **115**(8), 1690–1692 (2018)
16. Gogtay, N.J., Thatte, U.M.: Principles of correlation analysis. J. Assoc. Physicians India **65**(3), 78–81 (2017)
17. Chung, Y., et al.: Unknown examples & machine learning model generalization. arXiv preprint arXiv:1808.08294 (2018)
18. Prasad, A.M., Iverson, L.R., Liaw, A.: Newer classification and regression tree techniques: bagging and random forests for ecological prediction. Ecosystems **9**, 181–199 (2006)
19. Chen, J.-J., et al.: Identifying the top determinants of psychological resilience among community older adults during COVID-19 in Taiwan: a random forest approach. Mach. Learn. Appl. **14**, 100494 (2023)
20. ForouzeshNejad, A.A., Arabikhan, F., Aheleroff, S.: Optimizing project time and cost prediction using a hybrid XGBoost and simulated annealing algorithm. Machines **12**(12), 867 (2024)
21. Jakkula, V.: Tutorial on Support Vector Machine (SVM). School of EECS, Washington State University 37.2.5: 3 (2006)
22. Taud, H., Mas, J.-F.: Multilayer perceptron (MLP). In: Geomatic Approaches for Modeling Land Change Scenarios, pp. 451–455 (2018)
23. Panchal, G., et al.: Behaviour analysis of multilayer perceptrons with multiple hidden neurons and hidden layers. Int. J. Comput. Theory Eng. **3**(2), 332–337 (2011)

Automotive Accident Prevention System by Fuel and Electrical Circuit Deactivation

Dnyaneshwar Kanade, Aditya Inamdar, Suraj Gitte, Dev Jangam, and Rohan Humbe

Abstract This research proposes a real-time accident detection and mitigation system with the goal of reducing damage during vehicle crashes. The system uses an accelerometer sensor to track sudden changes in acceleration which may indicate an accident. When the system detects an acceleration rate that is higher than the set threshold, the system will activate a fuel cut-off device which will immediately block the engine's fuel pipes and stop the fuel supply. This rapid response improves the safety of the vehicle and its passengers by reducing the chance of fire and further damage. The recommended solution is an inexpensive standalone system that aims to improve current vehicle safety measures. This approach will help improve road safety and reduce the severity of the consequences of accidents by reducing damage and preventing harm after an accident.

Keywords Accident · Safety · Vehicle · Fuel tank

D. Kanade · A. Inamdar · S. Gitte · D. Jangam (✉) · R. Humbe
VIT Pune, Pune, India
e-mail: dev.jangam23@vit.edu

D. Kanade
e-mail: dnyaneshwar.kanade@vit.edu

A. Inamdar
e-mail: aditya.inamdar23@vit.edu

S. Gitte
e-mail: suraj.gitte23@vit.edu

R. Humbe
e-mail: rohan.humbe23@vit.edu

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_14

1 Introduction

Car accidents remain globally deadly and a major cause of injuries, underlining the immediate requirement of vehicle safety systems. While existing technologies such as automatic emergency braking and airbag systems reduce the effect, post-collision events such as fuel leaks, electrical failures, fires and fire hazards pose serious threats to first responders [1, 2].

This research introduces a cost-influential, real-time motor vehicle safety growth system that is designed to detect a collision using an accelerometer and immediately neutralize both fuel and electrical circuits [3, 4]. By integrating the ADXL345 accelerometer with Arduino UNO, the system monitors the acceleration pattern and begins a fuel shutoff response when an accident is detected. This action reduces the possibility of fire, fuel spillage, or second-charam accidents after an initial collision.

Unlike high cost or complex motor vehicle safety solutions, the proposed system is a Modu-Laar, standalone approach that can be easily embedded in existing vehicle architecture without wide engineering. The main contribution of this work lies in its simple yet effective mechanism to reduce the risk of post-collision, which makes it particularly suited for use in low-cost vehicles and developing areas.

2 Literature Review

Kassim et al. [1] analyzed acceleration sensors' performance in detecting vehicular motion anomalies, laying a foundation for real-time accident detection. However, their work stops at detection and does not propose post-accident mitigation strategies like fuel cut-off mechanisms.

Bala Aditya et al. [3] and Siam et al. [5] proposed sensor-based collision detection systems that communicate alerts to emergency services. While efficient in early warnings, they primarily rely on communication protocols, lacking mechanical intervention (e.g., deactivating fuel flow) as seen in our system.

Tahemeen and Patil [2], and Rakshith et al. [6], explored IoT-based accident response systems, suggesting smart fuel cut-off solutions. However, these systems are largely conceptual or depend on internet infrastructure, which can be unreliable in remote areas. Our system, by contrast, is a standalone embedded solution that operates even without external communication.

Baghdadi et al. [4] used Raspberry Pi and MEMS accelerometers for portable vibration monitoring, emphasizing flexibility. Our system offers a similar modular design but with the added functionality of mechanical intervention for accident mitigation.

Widianto et al. [7] and Rani et al. [8] explored behavioral tracking and vehicle motion analysis. While beneficial in accident prevention, our approach extends the safety boundary by addressing accident consequences via real-time deactivation of fuel and motor systems.

Maputi and Garlanka [9] analyzed structural design for crash resistance, supporting safer physical vehicle frameworks. Our work complements these findings by contributing an electronic solution that mitigates post-collision risks like fire.

Alsayaydeh et al. [10] and Gomathy [11] presented integrated systems using multiple sensors for crash detection and alert generation. Though comprehensive, their systems emphasize software communication and data analytics more than physical hazard mitigation. Our system uniquely prioritizes immediate physical action—cutting off the fuel supply—to directly reduce post-collision threats.

3 Methodology

The goal of this project is to create a multi-purpose automation system that uses an Arduino UNO microcontroller for efficient operation. The system connects necessary parts such as motors and fuel pumps. ADXL345 accelerometer sensor and other control modules This configuration is intended for applications that require acceleration monitoring. liquid pumping and speed control The system guarantees reliable operation without excessive power supply using a dedicated power source for the motor, fuel pump and main control unit [7]. Relay modules and user-controlled buttons increase flexibility by enabling the operation of automation or human automation components (Fig. 1).

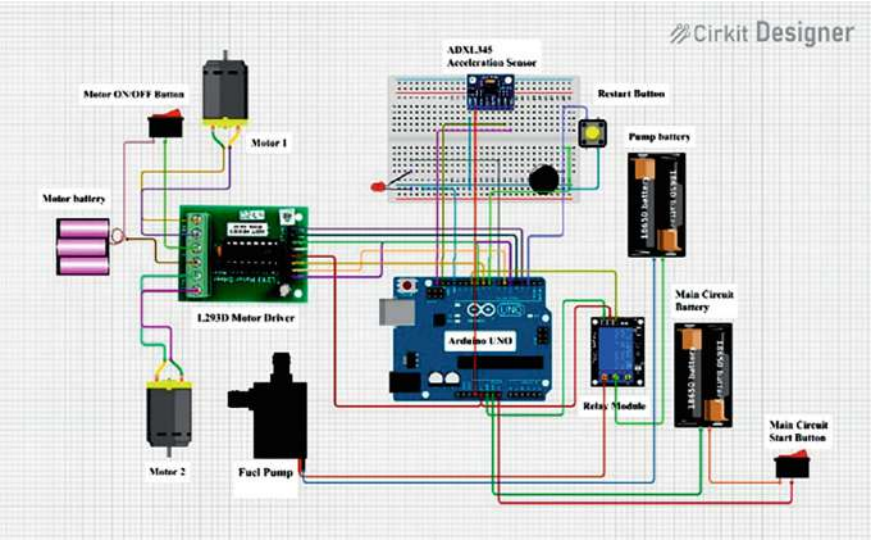


Fig. 1 Circuit diagram

The selection of components and overall system architecture was operated by the goals of ability, real-time accountability, hardware-level control and ease of integration. Each component performs an important function, which contributes to the reliability and simplicity of the system.

Arduino UNO: Chosen due to programming, low cost and ease of comprehensive community support. It provides adequate processing power for real-time sensor monitoring and control signal handling.

The ADXL345 Accelerometer: Selected for its 3-axis high-resolution output, compact form factor and low power consumption, which makes it ideal for sudden detection of vehicle acceleration changes [4, 12].

The L293D motor driver: Allows for accurate motor control and supports bidirectional current flowing, enabling responsible mechanical operations.

Relay Module: Controls the fuel pump's operation by acting as a switch and for safety, keeps high-power circuits away from the Arduino.

Fuel Pump: Turns on a liquid pump controlled using the relay module and the Arduino.

Batteries

(1) The motors are powered by the motor battery.
(2) The fuel pump is powered by the pump battery.
(3) Arduino and other control components are powered by the main circuit battery.

The other components used were: **5 V DC motors**, **buttons**, **5 mm LEDs**, etc.

A standalone embedded system design was preferred to ensure that the system remains functional in real time, even in the absence of external network or connectivity.

Compared to IOT infrastructure, GSM modules, or other relying more relying on complex microcontroller, this solution provides a practical trading between performance and practical viability, especially in cost-sensitive or resource-limited environment such as two-wheelers, small commercial vehicles, or in rural areas.

The main objective of this project is to demonstrate how various hardware components work together. Using the L293D motor driver and relay module, it interprets data from the accelerometer sensor and the Arduino UNO ADXL345 fuel pump to control the motor. This modular design provides a flexible answer for real-world automation needs, such as environmental monitoring. Liquid handling systems, robots, one of the applications can be changed according to category [6, 8].

Steps Involved:

1. **Accident detection:** The ADXL345 accelerometer sensor is used in the system to track variations in acceleration in order to identify accidents. The ADXL345 transmits data to the Arduino UNO for processing after measuring acceleration along the X, Y, and Z axes. To indicate anomalous or abrupt acceleration changes, such as those caused by a collision, a predetermined threshold value is set into

the Arduino. The system detects an accident when the acceleration value from any axis surpasses this level.

After detection, the Arduino can initiate certain functions, such sounding an alarm, halting the motors, or communicating with another device for emergency assistance. Because of this feature, the system can be used for industrial monitoring or automobile safety systems (Figs. 2 and 3).

2. **Blocking the fuel pump**: When an accident is detected, the fuel pump in the system is set up to shut off automatically. The ADXL345 accelerometer continually measures acceleration. The Arduino notifies the relay module in charge of the fuel pump if it receives signals indicating a sudden acceleration change that beyond the predetermined threshold, which would indicate an accident.

   When the relay module receives this signal, it functions as a switch and cuts off the pump's power supply. As seen in the circuit diagram, this guarantees that in the event of an accident, the pump will immediately stop operating to limit additional damage, dangers, or gasoline spills [6, 11, 13]. Particularly in applications such as automated fluid-handling systems or automobiles, this safety element adds an additional degree of protection.
3. **To restore default system**: A reset button on the system enables users to return it to its basic configuration in the event of an accident or unusual activity. The

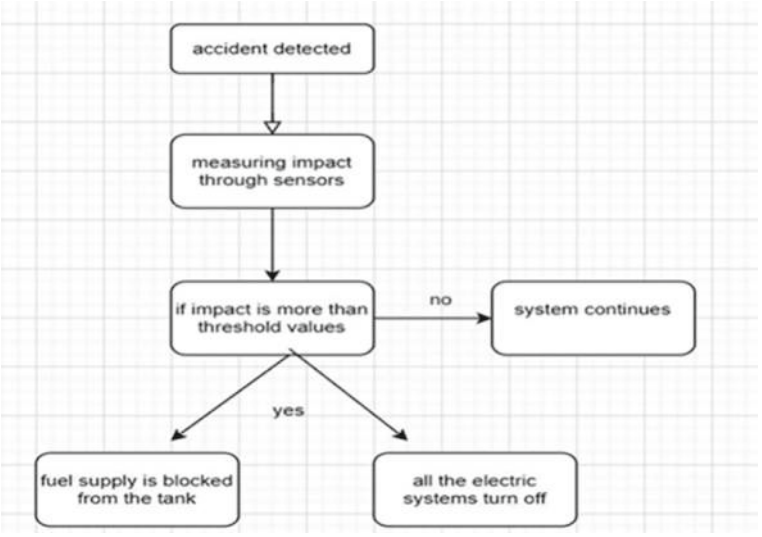


Fig. 2 System logical flow diagram

$$\text{Total acceleration (in g)} = \frac{\sqrt{(x^2 + y^2 + z^2)\ (\text{in m/s}^2)}}{g}$$

Fig. 3 Total acceleration equation

Arduino UNO receives a signal when the reset button is hit, clearing any error states or triggers brought on by the accident detection. By doing this, the system is reinitialized, enabling parts like the fuel pump and motors to function normally again. The reset feature makes the system easy to operate and perfect for situations that call for swift recovery by guaranteeing that it may be brought back online quickly and effectively without requiring human reconfiguration [10].

The below flow chart represents the whole process what will happen whenever the accident is detected.

4 Results and Discussions

Figure 4 shows the system before the accident detection.

Figure 5 shows the system when a collision is detected. The fuel pump and the dc motors connected to L293D motor driver are deactivated. The alert system (led and buzzer) is activated by the system. The system can be resumed manually by pressing the reset button.

Table 1 shows a sample data output of the Arduino IDE Serial monitor in tabular format. The total acceleration calculated by the microcontroller at every time stamp is compared with the predefined threshold value.

During the initial prototype construction, the threshold value was set to 1.5 g. The acceleration at time stamp 8 is greater than the threshold. Hence the system detected it as a collision and the fuel pump and dc motors were deactivated while the LED and buzzer were triggered. The system's ability to identify accidents and ensure efficiency and safety has been effectively proven. The Arduino UNO analyzes data

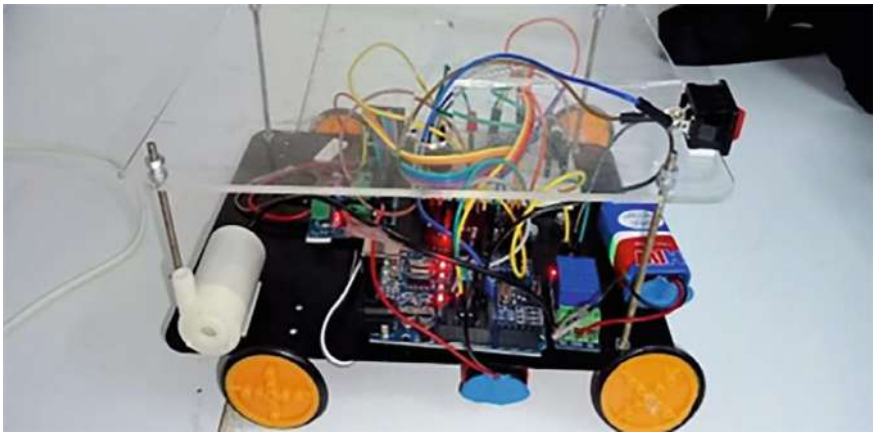


Fig. 4 Working of the system before collision detection

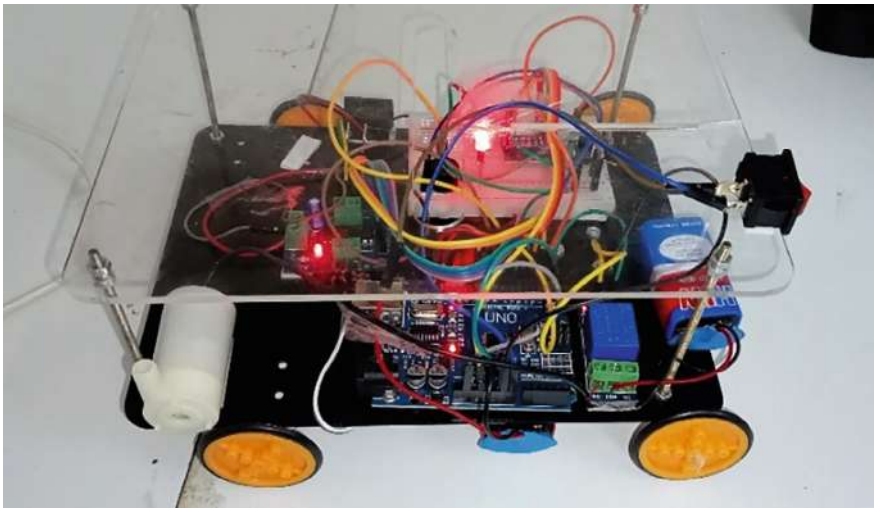


Fig. 5 System responding to a collision

Table 1 Readings in the Arduino IDE serial monitor before and after a collision

Time	X (g)	Y (g)	Z (g)	Total acceleration (g)	Collision detected
1	0.98	−0.31	−9.18	0.94	
2	0.94	−0.35	−9.34	0.96	
3	0	−0.43	−9.34	0.95	
4	−1.22	−0.24	−8.67	0.89	
5	6.94	0.35	−11.4	1.36	
6	−2.28	−0.78	−7.81	0.83	
7	−6.51	−0.27	−7.53	1.02	
8	8.16	2.31	−12.7	1.56	Collision detected
9	0.63	0.04	−8.75	0.89	
10	0.35	−0.51	−9.02	0.92	
11	0.39	0.27	−9.06	0.93	
12	−0.47	0.04	−9.34	0.95	
13	0.04	0.63	−9.65	0.99	
14	0.04	0.39	−9.45	0.97	

from the ADXL345 acceleration sensor, which accurately detects sudden changes in acceleration [5, 10].

The system also performs well in resuming operations after detecting a crash. All parts of the system including the pump and motor can work normally after pressing the reset button [10]. This feature ensures system reliability and flexibility for real-world applications. In addition, the power supply to the motors is divided. Fuel pump

and the main circuit guarantees continuous operation without overloading a single power source.

However, several limitations were noted during testing. For example, small vibrations or noise in acceleration data can mislead collision detection [12]. This can be reduced by improving the threshold level or by using sensor data filtering methods [4, 7]. Additionally, the inclusion of a wireless communication module in the actual system implementation improves its utility in automatic or remote setup by enabling Report or send an alarm in the event of an accident. Despite these minor disadvantages, the system remains applicable to industrial automation, robots, vehicles, safety and more. It is a possible solution for use.

The proposed system has been selected based on the critical need for real-time accident response mechanisms that extend beyond traditional detection and notification systems. Most existing solutions rely heavily on communication networks (e.g., GSM, IoT) or cloud-based data processing, which can introduce latency and are not always reliable in remote or underdeveloped regions. Our system addresses this limitation by offering a standalone embedded hardware solution that functions without external connectivity [2, 5].

This hardware-centric approach ensures immediate physical intervention-shutting off the fuel pump and motors upon detecting a collision through acceleration data. This significantly reduces the risk of fire, fuel leakage, or further mechanical damage after an accident, thereby improving overall vehicle and passenger safety.

Unlike prior work that primarily focuses on either structural improvements or communication-based notifications, our system emphasizes physical hazard mitigation by acting directly on the fuel and motor systems. The inclusion of such a mechanism is essential in reducing fatalities and secondary damage due to post-accident combustion or electrical hazards.

Additionally, the system uses low-cost components like the Arduino UNO and ADXL345 accelerometer, making it economically viable and accessible, especially in low-resource settings. Its modular design allows seamless integration into existing vehicle models with minimal structural modifications. It also lays the groundwork for potential future upgrades, such as GPS tracking or GSM module integration, without compromising the system's core offline functionality.

5 Future Scope

The future goals of this project are to improve practicality and increase efficiency. One possible improvement is the integration of wireless communication modules such as GSM, Wi-Fi or Bluetooth to facilitate remote monitoring and notification, for example in the event of an accident. The system can report its location and status to emergency services or designated contact with this functionality. The system will be more suitable for real-time monitoring applications in industrial automation or automotive safety. The benefits can be further improved by adding a GPS device, which provides precise location information during an accident.

Using cutting-edge sensor technology and machine learning algorithms is another way to improve. Multiple sensors, such as proximity or gyroscope it can help the system identify and classify different types of accidents or abnormalities. Machine learning can also be used to test for patterns in sensor data, reducing false positives and increasing the overall reliability of the system [12, 14].

6 Conclusion

This paper presents a low cost, design and implementation of a modular accident prevention system that automatically neutralizes the fuel of the vehicle on crash detection and electric circuit-composition. By taking advantage of the ADXL345 accelerometer and an Arduino-based control mechanism, the system can firmly identify sudden changes in acceleration and react in real time.

The main contribution of this work is the development of a practical and standalone safety solution that requires minimal vehicle modification [3, 7, 13], making it a viable option for extensive deployment in budget-friendly vehicles [15]. The system enhances existing passive security facilities, which reduces the threats after the collision, such as fuel leaks and fire [2, 11].

With wireless communication and extending GPS integration, the system has the ability to develop in a comprehensive post-cross reaction frame-work. The project highlights how accessible technology can be implemented effectively to address significant intervals in road safety.

References

1. Kassim, A.M., Jaya, A., Azahar, A.H., Jaafar, H.I., Sivarao, S., Jafar, F.A., Aras, M.S.M.: Performance analysis of acceleration sensor for movement detection in vehicle security system. Int. J. Adv. Comput. Sci. Appl. **10**(10), 1471 (2019)
2. Tahemeen, T., Patil, R.: IOT based solutions for accident detection and ıntimation. J. Sci. Res. Technol. **2**(9), 93–102 (2024)
3. Bala Aditya, D., Naresh, N., Vinay Kumar, K., Giri Raju, B.: Accident detection and alert system. J. Eng. Sci. **14**(06), 9254 (2023)
4. Baghdadi, H., Rhofir, K., Lamhamdi, M.: Smart portable system for monitoring vibration based on the Raspberry Pi microcomputer and the MEMS accelerometer. Int. J. Inform. Commun. Technol. **12**(3), 261–271 (2023)
5. Siam, S.M., Sumaiya, K.I., Al-Amin, M.R., Turj, T.H.: Automatic Motorbike Accident Detection and Notification System. BRAC University (2022)
6. Rakshith, M., Sanjana, K., Saikiran, T.R., Rushil, R., Prasad, R.K.: Smart fuel cut-off system for automobiles using IoT: a survey. In: Proceedings of the 2nd National Conference on Engineering Applications of Emerging Technology in Association with International Journal of Scientific Research in Science, Engineering and Technology
7. Widianto, E.D., Waskitaningrum, K., Isanto, R.: A Motorcycle Monitor and Control System for Teenager Riders. Department of Computer Engineering, Faculty of Engineering, Diponegoro University, Semarang (2017)

8. Rani, B., Sam, R.P., Kamatam, G.R.: A review on vehicle tracking and accident detection system using accelerometer. Int. J. Appl. Eng. Res. **13**(11), 9215–9217 (2018)
9. Maputi, E.S., Garlanka, S.B.: Evaluation of vehicle fuel tank ımpact resistance. Int. J. Sci. Res. **3**(7), 2319–7064 (2014)
10. Alsayaydeh, J.A.J., Yusof, M.F., Abdillah, M.A.A., Al-Gburi, A.J.A., Herawan, S.G., Oliinyk, A.: Enhancing vehicle safety: a comprehensive accident detection and alert system. Int. J. Adv. Comput. Sci. Appl. **14**(11), 5498 (2023)
11. Gomathy, C.K.: Accıdent detectıon and alert system. J. Eng. Comput. Archit. **12**(3), 7197 (2022)
12. Chen, Y., Wang, H., Liu, H.: AI-driven vehicle safety systems using embedded accelerometer data. Sensors **22**(3), 8220 (2022)
13. Li, K., Zhang, M., Zhou, Y.: A real-time fuel cut-off mechanism for electric vehicles post collision. J. Transp. Saf. Sec. **13**(2), 996 (2021)
14. Park, J., Lee, H.: Design of a smart car accident detection system with edge computing. IEEE Internet Things J. **10**(2), 4662 (2023)
15. Rakshith, R.M., Sanjana, K., Saikiran, T.R., Rushil, R.R., Prasad, R.K.: Smart fuel cut-off system for automobiles using IoT: a survey. Int. J. Sci. Res. Sci. Eng. Technol. **9**(4), 99 (2023)

Mental Health Assessment Using Machine Learning Models: A Comparative Review of Recent Advances

Kanupriya Arora and Kapil Joshi

Abstract Anxiety, depression, and stress are all psychiatric disorders that coexist and have an impact on the quality of life of people across the globe. They are influenced by environmental, psychological, and biological factors. Prognosis is important for early treatment and to minimize the effects of these diseases on the individual as well as on society. According to the WHO estimate, 1 out of every 8 people in the world have a mental illness. Severe impairment in the thought, emotional, or behavioural processes is a characteristic of mental diseases. Usually, wearable technology, social media activity, or self-reported questionnaires are used to gather data for stress, anxiety, and depression prediction. In order to identify patterns and risk factors for accurate prediction we generally use machine learning models and statistical techniques. The outcomes portray moderate to high effectiveness in depression, anxiety, and stress prediction, varying with the methodology and data quality. We have reviewed 13 different models for the prediction of stress, anxiety and depression. By comprehensive study we find out that neural network performed the best with the highest accuracy in terms of Accuracy, Error rate, Precision, Recall, F-measurer area.

Keywords Mental health · Artificial intelligence · Machine learning · Neutral network · Random Forest

1 Introduction

AI is revolutionizing several industries at a swift pace, and its impact on medicine has been so powerful. AI is transforming the way medical professionals treat patients by making clinical processes more efficient and directing decisions in the healthcare sector. The health care sector is presently collecting massive amounts of data from patients and hospitals. By leveraging this data effectively, physicians can predict

K. Arora (✉) · K. Joshi
Computer Science and Engineering Department, Uttaranchal University, Dehradun, India
e-mail: porwalkanupriya00@gmail.com

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*,
Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_15

more effective treatment approaches and enhance the overall delivery of healthcare services. The Python framework has emerged as a valuable tool in this endeavor, facilitating data analysis and computational processes that support informed decision-making. This research paper compares deeply and delves into the multifaceted contributions of AI to the life sciences, highlighting its potential to accelerate discoveries, improve patient outcomes and address pressing challenges in the field. We look at the state of artificial intelligence in the life sciences now, current research directions, and how AI might change healthcare in the future [1].

Fundamentally, AI is based on machine learning (ML), which gives computers the ability to learn from enormous volumes of data and forecast future events [2]. A kind of artificial intelligence called machine learning (ML) enables computers to learn from their experiences and get better without explicit programming. Since the fifth century BC, there have been references to mental illness, indicating that mental health difficulties have persisted throughout human history. However, because of their fast-paced lifestyles, many people in the modern world suffer from stress, anxiety, and despair, contributing to a notable increase in mental health issues.

In India, the topic of mental health remains stigmatized, resulting in inadequate healthcare support and a staggering number of people suffering from mental illness. According to statistics, 130 million people in India may be struggling with some form of mental health issue. The primary reasons behind this alarming number include a crumbling healthcare system and inadequate government support.

To combat this mental health epidemic, the authorities must take robust and essential steps towards healthcare, allocating sufficient funds towards mental health initiatives. A crucial aspect of this endeavour involves developing effective diagnostic tools, such as questionnaires that healthcare professionals can use to identify patients' conditions. Our study aims to forecast the following issues, leveraging AI and ML to develop innovative solutions that address the complex challenges in mental healthcare. Machine learning (ML)-based anxiety, depression, and stress prediction is gaining a lot of interest across all industries and is becoming an essential tool for bettering mental health care. Because of the stigma or lack of access, many people choose not to seek treatment for these disorders; therefore, ML-based systems can aid in the early detection of these problems [3].

1.1 Machine Learning Algorithm

The program or set of instructions which enables a system to learn from the existing data or information, identifying figures or patterns, and make predictions for each task with 0% program explicitly [4]. In essence, machine learning, or ML, is a branch of artificial intelligence (AI) that is primarily applied to the resolution of various issues in various fields. For instance, identifying novel patterns and insights in data, image recognition, natural language processing, and predictive analytics (Fig. 1).

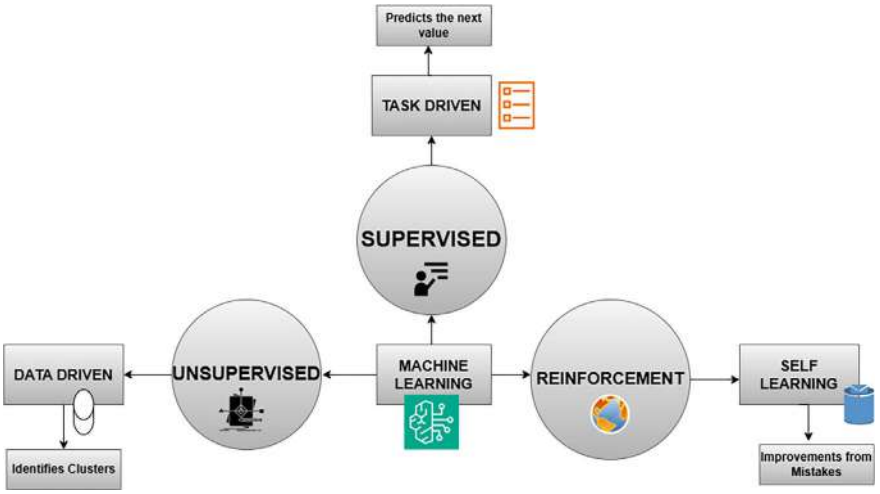


Fig. 1 Types of machine learning algorithms

1.2 Supervised Learning

In this learning is done from the existing knowledge and predictions is made on the new data. The labeled data are used by the algorithm for learning.

- For e.g.—teaching a child with the help of flash cards where the answers are already present.
- Predicting house prices based on size, location, etc.
- Email spam detection: Classifying emails as spam or not.

1.3 Unsupervised Learning

In this algorithm learning is done from data without labelled answers and finds hidden patterns or structures.

- For e.g.—giving a child a bunch of mixed items (toys, books, clothes) and asking them to organize them into groups.
- Grouping customers into segments based on shopping habits.
- Finding patterns in social media trends (Fig. 2).

1.4 Semi-supervised Learning

This method employs a combination of labelled and unlabelled data since it learns using both supervised and unstructured patterns.

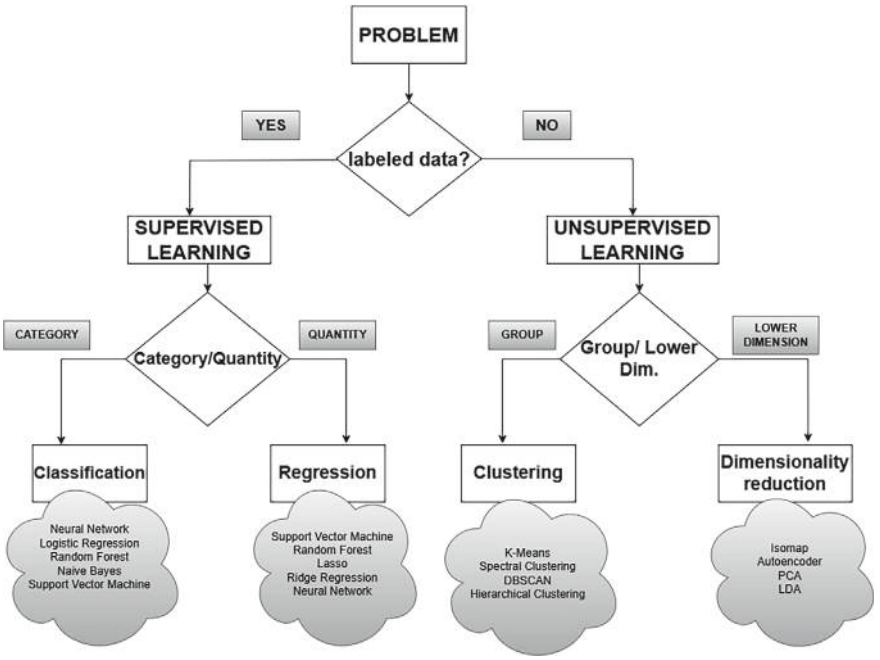


Fig. 2 Types of supervised and unsupervised

For e.g.: This is like giving a child some flashcards with answers and some without answers.

- Text classification where only a few documents are labelled.
- Fraud detection, where only some transactions are labelled as fraudulent.

1.5 Reinforcement Learning

Through interaction with the environment and feedback in the way of incentives or punishments, the algorithm gains knowledge.

For e.g.: teaching a dog a trick by rewarding it when it does the right thing.

- Self-driving cars: Learn to drive safely by practicing in a simulated environment.
- Game-playing AI: Like AlphaGo, which learns to play board games better than humans.

There are basically various steps involved to predict stress, anxiety and depression. We studied 31 research papers, out of them 28 are using machine learning algorithms few uses self-report questionnaires, few uses clinical interviews, and few uses physiological measurements using wearable sensors. We have reviewed 13 different models for the prediction of stress, anxiety and depression. By comprehensive study we find

out that Neural network performed the best with the highest accuracy of 99.9% in terms of Accuracy, Error rate, Precision, Recall, F-measurer area.

2 Related Works

This study explores the use of machine learning (ML) algorithms to identify predictors of depression using data from the National Health and Nutrition Examination Survey (NHANES) 2017–2020. The dataset includes medical, mental, demographic, and lifestyle information from 8965 individuals aged 18 to 80 years. Seven ML algorithms were tested, with the Neural Network algorithm achieving the highest performance, indicated by an area under the curve (AUC) of 91.34%. This performance significantly outperformed traditional statistical methods such as logistic regression [3, 5, 6]. Key findings include that age, health, childhood conditions, and low education levels are significant predictors of depression risk in later life. New prognostic patterns are also found, including low dental care utilization and life course instability. The study employs the Shapley Additive Explanations (SHAP) method to elucidate these predictive patterns [7, 8]. The research paper explores the critical role of mental health, which encompasses emotional, psychological, and social well-being, in shaping our thoughts, emotions, behaviours, responses to stress, interactions with others, and decision-making processes. The paper examines different machine learning models, techniques, and applications in this field, with an emphasis on data modalities, in recognition of the growing interest in applying machine learning for the early identification of mental illness. From a technical standpoint, standard logistic models are consistently outperformed by more sophisticated SML algorithms. However, predictive performance is greatly improved when a Gradient Boosting model is used in conjunction with semi-structured input data depending on life sequence attributes. According to the study, structured input data enhances interpretability and predictive accuracy; the most dependable PR-AUC, given the information at hand, is 0.77 for females and 0.65 for males [9–11]. The document details a study on predicting personality traits and stress levels using various machine learning models. The traits analyzed include extroversion/introversion and neuroticism/stability, which are used to estimate stress levels. Data is collected through a questionnaire, and multiple machine learning classifiers are applied to this data [2, 12]. The document provides a comprehensive review of AI algorithms used for stress prediction based on heart rate variability (HRV) data. It discusses various rule-based (RB), shallow machine learning (ML), and deep machine learning (DML) approaches, highlighting their respective advantages and challenges. The review emphasizes the importance of accurate stress prediction for mental health and its impact on socioeconomic conditions. It compares different methodologies, datasets, and performance metrics used in recent studies, presenting detailed tables summarizing the results. The document concludes with discussions on the challenges faced in this research area, such as data quality, biased datasets, and the need for real-time stress monitoring systems [13–15]. The review highlights the potential of ML in

mental health diagnostics, emphasizing the need for standardized methods and more research in clinical settings (Table 1) [1].

3 Comparative Analysis

As shown in Fig. 3, the raw data is the unprocessed information that is obtained from users directly without any modification. It comprises various formats like numerical values, text inputs, and possibly multimedia information like images and audio files. Such a complete set of data seeks to reflect various behavioural and cognitive factors pertinent to psychological condition analysis and categorization. Before applying models, the data that has been gathered goes through several pre-processing steps. These are cleaning of the data to eliminate inconsistencies or missing values, normalization to scale all the features to an equivalent scale, and transformation processes like tokenization or encoding in order to be compatible with machine learning algorithms. The goal here is to transform raw data into a structured form that can be analyzed while still maintaining its intrinsic properties. Once pre-processed, there are several machine learning models formed to study patterns and correlations across the dataset. These models involve Logistic Regression, Support Vector Machines (SVM), Random Forest, XGBoost, and Neural Networks. Every model has been set with hyperparameters set in an optimum state to drive better prediction as well as the main goal of predicting psychological disorders with high accuracy. The dataset is divided into two subsets: a training set for the models to be trained on and a testing set for the models to be tested on their prediction abilities. In training, the models learn the internal patterns from labeled data, and in testing, their ability to generalize is found out. The process ensures that the models are not only fit to know data but are also able to handle unseen data correctly.

Performance of the models is measured in terms of various performance indicators such as accuracy, speed, user satisfaction, cost-effectiveness, and processing time. Disorder-specific accuracy is also measured to realize the effectiveness of each model in identifying different psychological disorders like Depression, Anxiety, PTSD, OCD, and Bipolar disorder. All these measures give an overall picture of model reliability and feasibility.

4 Results Analysis

Each model's performance is measured quantitatively along various parameters: accuracy, time taken for execution, user satisfaction, cost for 100 runs, and speed per sample. As shown in Fig. 4, Neural Networks have the best accuracy (95%) and user satisfaction (9.5/10) compared to all other models, but with increased computational cost and time (30 s and $0.4 for 100 runs). XGBoost is also found to have strong performance with 93% accuracy, lowest cost, and highest speed and hence

Table 1 Comparative analysis

	Algorithm used	Model complexity	Suitable data size	Result	Accuracy	Performance metric
1	Decision Tree (DT), Random Forest Tree (RFT), Naïve Bayes (NB), Support Vector, Machine (SVM) and K-Nearest Neighbour (KNN)	Low to medium	Low to medium	Naïve Bayes	0.8555	F1 measure Score
2	Naïve Bayes (NB), Bayesian Network (BN), K-Nearest Neighbor (KNN), K-Star (a variant of KNN using entropy distance), Multilayer Perceptron (MLP), Radial Basis Function Network (RBFN), J48 (Decision Tree), Random Forest (RF), Hybrid of K-Star and Random Forest	High	Medium	Neural Network (radial basis function network)	0.999	Accuracy, Error rate, Precision, Recall, F-measurer area
3	Random Forest Algorithm	High	Medium	Random Forest model	0.802	Area under the receiver operating characteristic curve (AUC)

(continued)

Table 1 (continued)

	Algorithm used	Model complexity	Suitable data size	Result	Accuracy	Performance metric
4	Least Absolute Shrinkage Selection Operator (LASSO), K Nearest Neighbors (k-NN), Support Vector Machine Kernel (SVM-K), X Gradient Boosting (XGB), Random Forests, Naive Bayes Logistic Regression	High	Medium	Random Forests	0.921	Accuracy, Area under the curve (AUC), Positive predictive value (PPV), Recall, F-measure
5	Neural Networks, Logistic Regression, Decision Trees, Random Forests, Support Vector Machines (SVM), Nearest Neighbors (KNN), Naive Bayes	High	Medium	Neural Network algorithm	91.34	
6	Support Vector Machines (SVM), Neural Networks, K-Nearest Neighbors (KNN), Reinforcement Learning	High	No data was used	ML	99.80%	
7	K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Adaptive Boosting (AdaBoost), Random Forest (RF)	Medium	Medium	Support Vector Machine (SVM)	94.33%, AUC = 0.937	F1-score

(continued)

Table 1 (continued)

	Algorithm used	Model complexity	Suitable data size	Result	Accuracy	Performance metric
8	Decision Tree, Random Forest, AdaBoost, Gaussian Naive Bayes, Support Vector Machine (SVM), Fuzzy Pattern Classifier, Multi-Model Evolutionary Classifier, Fuzzy Pattern Classifier with Genetic Algorithm, Fuzzy Reduction Rule Classifier	High	Medium	Support Vector Machine (SVM)	91.43%	
9	CatBoost, Logistic Regression, Naïve Bayes, Random Forest, Support Vector Machine (SVM)	Medium	Medium	CatBoost	82.6%	Accuracy, precision, and AUC of ROC

(continued)

Table 1 (continued)

	Algorithm used	Model complexity	Suitable data size	Result	Accuracy	Performance metric
10	Penalized Linear Regression (Ridge, LASSO, Elastic Net) Decision Trees (Bayesian additive regression trees, Random Forest), Deep Learning Models (e.g., neural networks), Support Vector Machines (SVM), Clustering Algorithms (e.g., K-means, hierarchical clustering) Gradient Boosting Machines	High	Medium	Neural Network	82%	Accuracy, Precision, Area Under the Receiver Operating Characteristic Curve (AUROC)
11	Adversarial SID-loss maximization (ADV), SID-loss equalization with variance (LEV), SID-loss equalization using Cross-Entropy (LECE), SID-loss equalization using KL divergence (LEKLD)	High	Medium	LECE	80%	F1-Score, Gain in Voice Distinctiveness (GVD), De-Identification Score (DeID)

(continued)

Table 1 (continued)

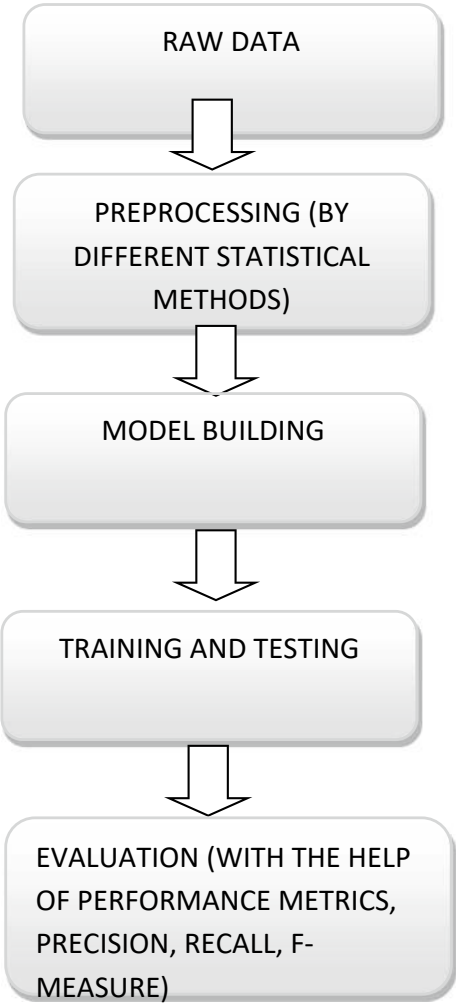
	Algorithm used	Model complexity	Suitable data size	Result	Accuracy	Performance metric
12	Custom CNN-RNN, MobileNetV2-RNN InceptionV3-RNN	High	Medium	InceptionV3-RNN	66%	Accuracy, Precision, Recall, F1 Score
13	XGBoost (Extreme Gradient Boosting)	High	High	XGBoost	92.8%	Area Under the Curve (AUC): Area Under Precision-Recall Curve (AUPRC), Sensitivity, Specificity

(continued)

Table 1 (continued)

	Algorithm used	Model complexity	Suitable data size	Result	Accuracy	Performance metric
14	Rule-Based Approaches: Fuzzy Logic (FL), Fuzzy Neural Networks (FNN), Fuzzy Adaptive Resonance Theory (ART), etc Shallow Machine Learning: Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Decision Trees (DT), Naive Bayes (NB), Random Forest (RF), Logistic Regression (LR), etc Deep Machine Learning: Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, Backpropagation Neural Networks (BPNN), Artificial Neural Networks (ANN), etc	Rule-based approaches: Low Shallow machine learning: Medium Deep machine learning: High	Medium to high	1. Shallow ML 2. Deep ML	Shallow ML—99.1% Deep ML—96.4%	Accuracy (Acc) Area Under the Curve (AUC) Precision (Pre) Recall (Rec) Sensitivity (Sen) Specificity (Spe) False Acceptance Rate (FAR) False Rejection Rate (FRR)
15	One-Sample t-Test; Multiple Regression Analysis	Low to medium	Low to medium	Multiple regression		Adjusted R Square; F Ratio; Significance Levels (p-values; t-values)

Fig. 3 Proposed workflow



is a well-balanced option for real-time usage. In addition, disorder-specific analysis as shown in Fig. 5 indicates that Neural Networks always provide the highest prediction scores for all categories with a maximum accuracy rate of 93% for the detection of depression. XGBoost and Random Forest follow very closely, with very high scores for disorders such as PTSD and Anxiety. Logistic Regression, although most efficient and fastest, is behind in the accuracy of performance compared to deep models. These results reinforce the balance between prediction accuracy and computational cost, revealing lessons concerning model choice given the needs of particular applications.

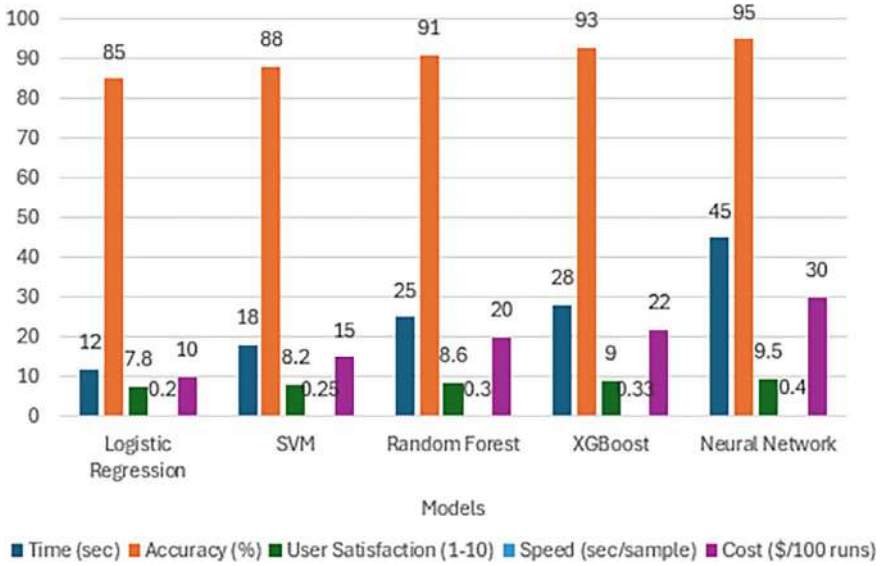


Fig. 4 Performance metrics comparison

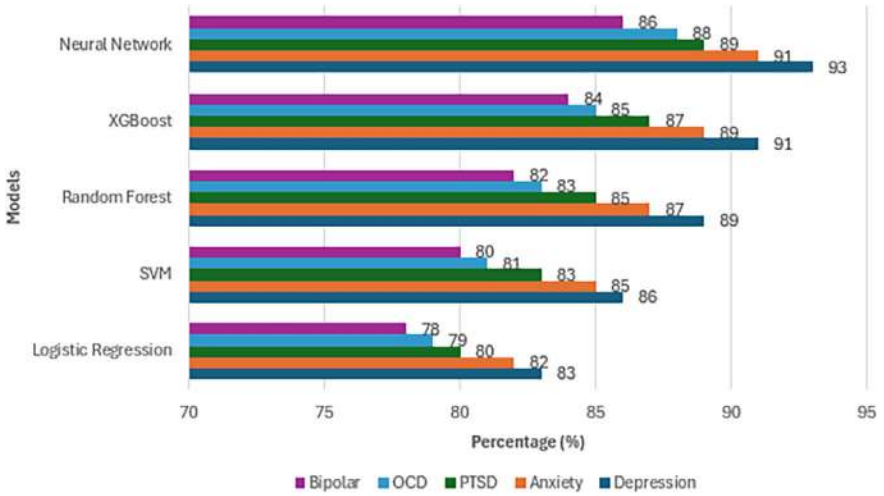


Fig. 5 Precision comparison

5 Conclusion

Numerous significant inferences are drawn from the study's findings. The validity of the questionnaire and the elements influencing the outcomes that are intended to be examined are well-founded in the reliability and precision of the studies. Thirteen

distinct models have been used to predict stress, anxiety, and depression. With the maximum reliability of 99.9% in terms of precision, error rate, recall, precision, and F-measurer area, the neural network outperformed the others. The technology and engineering department is not the only department to which this research can be expanded in the future.

References

1. World Health Organization: Mental Disorders. WHO, Geneva (2022)
2. Krishna, R., Teja, R., Neelima, N., Peddi, N.: Advanced machine learning models for depression level categorization using DSM 5 and personality traits. Proc. Comput. Sci. **235**, 2783–2792 (2024)
3. Jamali, A.A., Berger, C., Spiteri, R.J.: Identification of depression predictors from standard health surveys using machine learning. Curr. Res. Behav. Sci. **7**, 100157 (2024)
4. Ciharova, M., Amarti, K., van Breda, W., Peng, X., Lorente-Català, R., Funk, B., et al.: Use of machine-learning algorithms based on text, audio and video data in the prediction of anxiety and post-traumatic stress in general and clinical populations: a systematic review. Biol. Psychiatry (2024)
5. Canadian Mental Health Association. Connection Between Mental and Physical Health. Canadian Mental Health Association: Mental Health for All (2016)
6. Bertie, L.A., McDermott, E.A., Quiroz, J.C., Hudson, J.L.: A scoping review of clinical decision support systems for child and adolescent mental health. Curr. Psychol. **44**, 2785–2804 (2025)
7. Montorsi, C., Fusco, A., Van Kerm, P., Bordas, S.P.: Predicting depression in old age: combining life course data with machine learning. Econ. Hum. Biol. **52**, 101331 (2024)
8. Hennekens, C.H., et al.: Schizophrenia and increased risks of cardiovascular disease. Am. Heart J. **150**(6), 1115–1121 (2005)
9. Islam, M.M., Hassan, S., Akter, S., Jibon, F.A., Sahidullah, M.: A comprehensive review of predictive analytics models for mental illness using machine learning algorithms. Healthcare Anal. **6**, 100350 (2024)
10. Freeman, M.: Investing for population mental health in low and middle income countries—where and why? Int. J. Ment. Health Syst. **16**(1), 1–9 (2022)
11. Shao, T., et al.: Physical activity and nutritional influence on immune function: an important strategy to improve immunity and health status. Front. Physiol. **12**, 1702 (2021)
12. Kang, H.-J., et al.: Impact of anxiety and depression on physical health condition and disability in an elderly Korean population. Psychiatry Investig. **14**(3), 240 (2017)
13. Haghish, E.F., Czajkowski, N.: Reconsidering false positives in machine learning binary classification models of suicidal behavior. Curr. Psychol. **43**(11), 10117–10121 (2024)
14. Patel, V.: Mental health in low-and middle-income countries. Br. Med. Bull. **81**(1), 81–96 (2007)
15. Hasan, M.K., Zannat, Z., Shoib, S.: Mental health challenges in Bangladesh and the way forwards. Annals Med. Surg. **80**, 104342 (2022)

DigiDine: Digital Menu Card and Restaurant Ordering System

Ram Joshi, Tejashri Adsure, Ajay Dhakane, Sumit Karanjkar, and Ajay Kamble

Abstract The DigiDine is a web application designed to enhance restaurant operations using modern technology. Built using the MERN stack, this application provides a customizable digital menu that allows customers to directly order from their devices by scanning a table-specific QR code. The platform streamlines the dining experience by integrating a payment interface that facilitates seamless bill generation and payment processing. The app also includes a kitchen management interface, allowing staff to efficiently track and prepare meals based on real-time orders while maintaining oversight of inventory levels. By capturing user data, the platform enables restaurants to conduct data analysis, gaining valuable insights into customer behavior for service improvement. In addition to in-restaurant services, the app supports pre-order functionality, allowing customers to place orders before arriving, thus reducing wait times. Though currently limited in capacity, the system is designed for scalability, supporting the future growth of multiple restaurants. Planned enhancements include venue booking features, expanded payment options, and advanced data analytics for more personalized customer recommendations.

Keywords QR code · Application · Digitization · Data analysis · Reservation

1 Introduction

The restaurant industry is undergoing significant transformation as digital solutions replace conventional methods to improve service delivery and customer interaction. Printed menu cards, once a staple of the dining experience, are now seen as inefficient due to their static nature and the difficulty involved in updating them regularly.

R. Joshi · T. Adsure · A. Dhakane · S. Karanjkar (✉) · A. Kamble
Department of Computer Engineering, JSPM's RSCOE, Tathawade, Pune, India
e-mail: sumitkaranjkar741@gmail.com

R. Joshi
e-mail: rbjoshi_it@jspmrscoe.edu.in

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_16

To address these limitations, this paper introduces DigiDine, a web-based application developed using the MERN stack. The system allows restaurant patrons to access a digital menu by scanning a unique QR code linked to their table, place their orders directly from their devices, and make secure payments either before or after dining. It further supports pre-order functionality, enabling customers to reserve meals and tables ahead of time, thereby minimizing wait periods.

The principal innovation of this work lies in the integration of end-to-end restaurant services—from digital menu access and ordering to real-time kitchen coordination, payment processing, and behavior-based user recommendations. The system also supports multi-restaurant scalability and is designed with future enhancements in mind, such as event booking and advanced analytics, making it a comprehensive platform for modern restaurant management.

2 Literature Survey

In recent years, several digital systems have been proposed and implemented to improve restaurant and canteen services through automation, digitization, and intelligent features. However, many of these solutions suffer from critical limitations, such as limited scalability, dependency on specialized hardware, or lack of real-time integration between users, management, and kitchens. This section reviews and contrasts these existing systems while establishing the unique contributions of the proposed DigiDine solution.

An AI-driven cross-platform system described by Raibagi et al. [1] focuses on reducing canteen workload by automating order placement and integrating wallet-based payments. The system includes a digital menu and a feedback module but lacks support for in-restaurant ordering, as orders are routed through an admin first. This sequential processing introduces potential delays and reduces real-time kitchen integration.

Ahmed and Taj Kiran [2] proposed a ZigBee and RFID-based wireless ordering system where customers use touchscreens to order food. While ZigBee ensures low-power wireless communication, the system requires complex hardware components including LCDs and RFIDs. This makes the setup costly and difficult to maintain. Moreover, the architecture supports only a single restaurant, limiting its scope.

Gunawardena et al. [3] introduced a deep learning-powered digital menu system that supports ingredient insights, nutrition facts, dish preparation videos, and language preferences. Personalized recommendations are generated using customer history and registration data. Despite its innovation in health personalization, the system has limitations related to food item traceability and does not support group ordering or collaborative dining scenarios.

Lambora and Gupta [4] developed a smart ordering system using Android tablets over a shared Wi-Fi connection. The system enables food selection through images and introduces a “table replacement” concept. However, it does not support bill

payments from the tablet and requires mandatory user registration before order placement, which adds friction to the user experience.

The v-Canteen system [5] enables customers to place online food orders via a mobile app, minimizing wait times. It combines AI tools and hardware for crowd estimation and personalized notifications. However, the prototype has limited replicability across different locations and universities, making it unsuitable for broader deployment.

Ravi et al. [6] designed an Android-based touch interface placed at each table for order selection using Bluetooth. The system uses a thermal printer for bill generation and relies on microcontrollers to notify kitchen staff. Nevertheless, it lacks remote table booking and is limited by Bluetooth's range, hindering usability in larger spaces.

El Fiorenza et al. [7] suggested a simple digital menu browsing app that eliminates printed menu cards and reduces staff workload. Recommendations are based on customer history, and standard web technologies like databases and APIs were employed. However, this system lacks advanced features such as table-specific ordering or admin controls.

Liyanage et al. [8] developed a smart restaurant app using customer social media data for menu personalization and used sentiment analysis to summarize reviews. While users can view table status using Google Maps, the system lacks online payment options and features like table replacement or real-time kitchen updates.

Pieska et al. [9] outlined an "intelligent restaurant" setup consisting of four applications and service robots. Tablets display menus with nutrition and calorie info. Orders go directly to the kitchen and are marked complete post-preparation. Despite technological novelty, the system lacks modularity, extensibility, and multi-restaurant capabilities.

Samsudin et al. [10] introduced a customizable food ordering system with real-time customer feedback. Orders are placed using smartphones, and receipts are shared via private logins. Although practical, the system is restricted to single restaurants and lacks predictive analytics and table-specific workflows.

Yang et al. [11] proposed LAMF, a lighting control and ordering system using RFID. The system adjusts table lighting based on the selected food. However, its reordering logic is inefficient, requiring users to restart the process for new orders. Additionally, the RFID setup introduces bulk-ordering vulnerabilities due to tag reuse.

Mishra et al. [12] introduced a GSM/CDMA-based system for managing orders through Android devices on each table. The system enables restaurant managers to monitor orders and feedback, but future improvements are needed for online payments and off-premise orders.

Lin et al. [13] developed an NFC-based restaurant ordering application using P2P communication and membership card rewards. While it adds gamification through bonuses and coupons, the app lacks integration with table management or kitchen workflows.

Harpanahalli et al. [14] proposed an RFID-based smart restaurant system using Python and Raspberry Pi. This reduces time spent on ordering and payments.

However, it requires dedicated hardware and may face limitations in mass adoption due to technical overhead.

Domokos et al. [15] created Netfood, a software platform supporting multiple restaurant deliveries using a centralized backend and MVC architecture. However, it primarily supports delivery services and lacks dine-in specific features such as QR-based ordering or admin panel customization.

Rawat et al. [16] designed a voice-command-enabled table ordering system, connecting backend servers and tablets via Wi-Fi. It supports real-time order tracking and UPI integration, but its reliance on Wi-Fi security and lack of personalization modules restricts its utility.

Guiling and Qingqing [17] presented a self-service ordering system using ZigBee, promoting energy efficiency and wireless data transfer. While suitable for basic functionality, the system is limited in terms of data analytics, scalability, and integration with advanced user interfaces.

Liyanage et al. [18] again proposed Foody, which supports 3D menu modeling, sentiment analysis, and ingredient tracking. It solves various restaurant operation problems but doesn't directly address data security or real-time admin controls.

Hongzhen et al. [19] developed a web services-based food ordering system with mobile and PC support. The orders are transmitted to context and web servers. While efficient in structure, the system lacks features like home delivery, payment integration, and table reservation.

Lin et al. [20] reiterated NFC-based food ordering apps, focusing on user tracking and reward distribution. However, they omit essential functions like online payments and personalized recommendations.

Gupta and Saxena [21] implemented a wireless menu card for in-restaurant automation via tablets. Despite real-time updates and improved order accuracy, it faces scalability issues and relies heavily on stable internet connections.

Albawi et al. [22] offer foundational understanding of CNNs which are relevant to some food recognition or recommendation systems, though not specific to restaurant automation.

Comparison with DigiDine

Unlike many of the reviewed systems, **DigiDine** is a comprehensive, web-based platform using the **MERN stack**. It supports:

- **QR-based, table-specific ordering**,
- **Admin-managed dynamic menus**,
- **Real-time kitchen dashboards**,
- **Secure pre/post-dining payments**,
- **Multi-restaurant support**,
- **Pre-booking and reservation features**, and
- **Personalized recommendations** using machine learning techniques like collaborative and content-based filtering.

Its **modular architecture** enables future scalability and integration with features like venue booking, expanded payment systems, and customer behavior analytics, positioning DigiDine as a next-generation digital dining solution.

3 Methodology

DigiDine system has a comprehensive and systematic architecture. The architecture shows the proper steps involved in the overall system workflow. The system has two distinct modules, as shown in Fig. 1.

System Architecture

Users about to reach the restaurant will access the website and add dishes to the cart. Then it is followed by payment and bill generation. Users who reach in restaurant and want to place an order need to scan the QR code then he/she will get table table-specific menu page. Then the user will order the item of their choice.

Restaurant admin is able to edit menu cards as per food availability. Admin handles the payment module. The orders were then finally sent to the kitchen staff. The kitchen staff is able to access the order for a particular user.

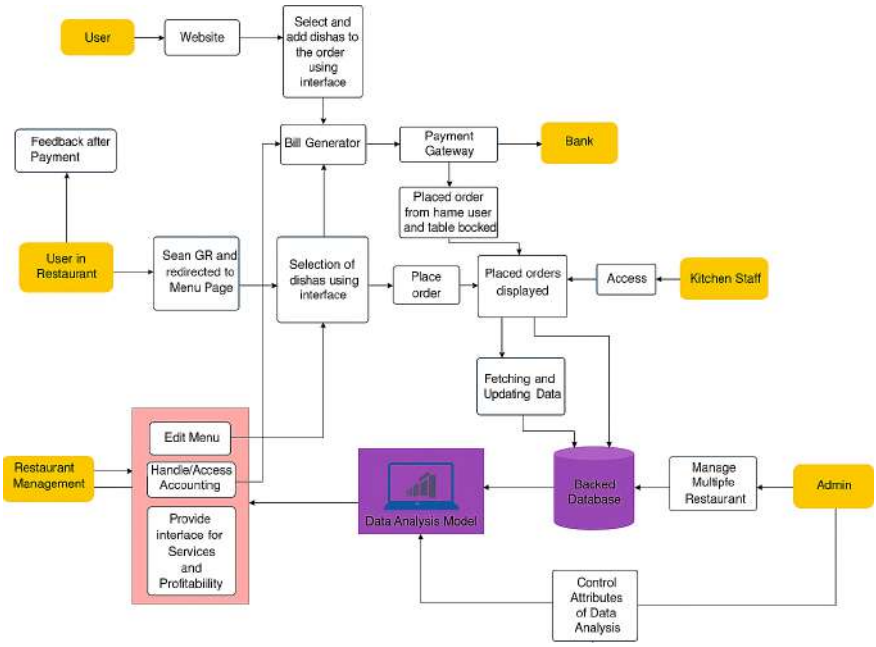


Fig. 1 Proposed workflow

Admin of the DigiDine system is able to handle multiple restaurants that are using this service. On the DigiDine website, multiple hotel menus are available, users need to select a hotel accordingly. Particular restaurant management can edit their menus accordingly.

Web Development Methodology

The MERN stack enables the creation of a dynamic, user-friendly web application that facilitates seamless interaction between restaurant customers and the ordering system. It ensures smooth data exchange for efficient menu browsing, order placement, and real-time updates, enhancing the overall dining experience.

Tech-Stack

MongoDB: MongoDB is used to store data in a document-oriented and flexible format. It is a NoSQL database. It has flexibility in storing unstructured data. It has high availability and scalability with replica sets and sharding. It has JSON-like document storage that fits with the data format used by JavaScript (BSON format).

Express.js: Express.js is a backend framework. It is a minimal and flexible Node.js web application framework that provides robust features for building web and mobile applications. Express.js acts as a middle layer between the database and the front end. Express.js simplifies the routing. It acts as middleware support for handling requests and responses.

React.js: It is a JavaScript library. It is used for building user interfaces, mainly for single-page applications. React.js has a component-based architecture. It consists of efficient updates through a virtual DOM, ensuring a smooth user experience.

Node.js: Node.js is used to build the backend of the application. It allows running JavaScript on the server side. It enables handling requests between the front end and the database. It has an asynchronous, event-driven architecture for high performance. It has a large ecosystem of libraries and modules through NPM.

Predictive Model and Data Analysis

The prediction and data analysis modules are integrated into the system to analyze user behavior by considering past data and providing recommendations to the user.

Machine Learning Integration

Machine learning algorithms used for personalized product recommendations are:

(1) Collaborative filtering.
(2) Content-based filtering.

Collaborative filtering: This approach gives recommendations based on user behavior.

(a) User-based collaborative filtering: Users having the same interests are segregated and items liked by users in the same category are recommended to one another.

(b) Item-based collaborative filtering: Items are grouped as per the ratings. According to that items are recommended to the user.

Content-based filtering: In this, items are recommended to a user based on what he likes previously and here important factor is an attribute of the item. Different attributes of an item are taken into account and based on the attribute items are recommended.

Novelty of the Proposed Approach

The proposed **DigiDine** system introduces a novel, end-to-end digital restaurant management platform that seamlessly integrates **customer interaction, administrative control, and kitchen coordination** into a single, unified solution. Unlike existing systems that typically address one or two functional areas—such as digital menus, feedback collection, or wireless ordering—DigiDine offers a **holistic ecosystem** that digitizes the entire dining experience while maintaining **scalability, personalization, and operational efficiency**.

1. **Table-Specific, QR-Based Ordering**

One of the standout innovations in DigiDine is the implementation of **table-specific QR codes**. Upon scanning the QR, customers are directed to a dynamically generated digital menu that corresponds to their specific table. This reduces ordering errors, eliminates the need for dedicated hardware at each table, and enhances the user experience by allowing customers to order directly from their personal devices.

2. **Dual-Mode Ordering and Payment System**

The system uniquely supports both **in-restaurant** and **pre-arrival** ordering modes. Customers can place and pay for their meals before reaching the restaurant, enabling faster service and reduced wait times. Alternatively, they can order and pay after arriving and finishing their meal, providing flexibility and convenience that many existing systems lack.

3. **Multi-restaurant Support with Centralized Admin Management**

Unlike previous solutions that are often restricted to a single venue, DigiDine is designed to accommodate **multiple restaurants on a shared platform**. Each restaurant's admin panel allows for **menu customization, order tracking, and inventory control**, all accessible via a centralized interface. This architecture enables the system to scale across franchises or partner outlets with ease.

4. **Real-Time Kitchen Interface**

The system provides kitchen staff with a **live order dashboard**, streamlining communication between the front-end and back-end. Orders are categorized and timestamped, allowing for efficient meal preparation and prioritization. This minimizes delays, reduces manual miscommunication, and ensures that customer expectations for service speed are met.

5. **Integrated Machine Learning for Personalized Recommendations**

Another novel feature is the integration of **predictive analytics and recommendation engines**. Using collaborative and content-based filtering techniques, the system learns user preferences based on order history and menu interactions. This enables tailored menu suggestions, enhancing customer engagement and satisfaction.

6. **Planned Expansion Capabilities**

The system is designed with **future extensibility** in mind. Upcoming features such as **venue booking**, **event management**, and **advanced analytics dashboards** are already supported by the current architecture, showcasing its readiness for evolution beyond standard restaurant functions.

Summary of Novel Contributions

- Combination of **QR-based, table-specific interaction** with **multi-mode ordering and payment**.
- A **centralized system** that supports **multi-restaurant operations** through modular admin panels.
- Inclusion of **real-time kitchen workflow management** synchronized with frontend activity.
- Use of **machine learning** to provide **dynamic and personalized user recommendations**.
- **Pre-ordering and table reservation** capabilities that reduce wait time and improve operational flow.
- **Scalable design** capable of integrating future services like event bookings, promotions, and cross-platform analytics.

Together, these innovations distinguish DigiDine from existing food ordering systems by delivering a truly integrated, data-driven, and scalable digital restaurant experience.

4 Results

The system workflow is as given below:

(1) The user will sign up in the system if he/she is a new user (Fig. 2). If the user already exists in the system then only needs to log in.
(2) The user can explore the menu of the restaurant. Various categories of food are shown with their images (Fig. 3).
(3) The user needs to reserve the table (Fig. 4). To make a reservation user needs to fill in basic details like name, date, etc.
(4) All the food items purchased by the user are added to the cart and then the order summary is shown (Fig. 5).

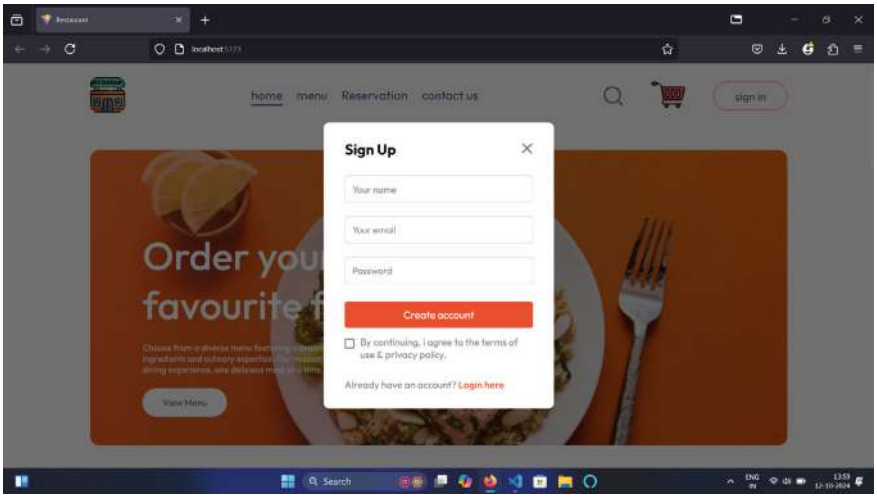


Fig. 2 Developed website sign-up page

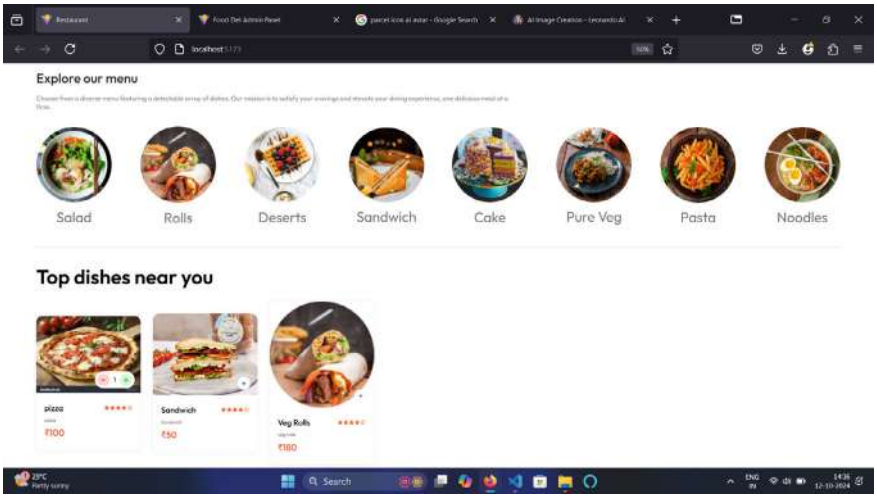


Fig. 3 Customized menu

(5) An admin panel is provided where the admin can edit the menu of the website. Admin also checks the status of an order whether it is in preparation or delivered state (Fig. 6).
(6) The admin panel shows the placed order summary (Fig. 7).

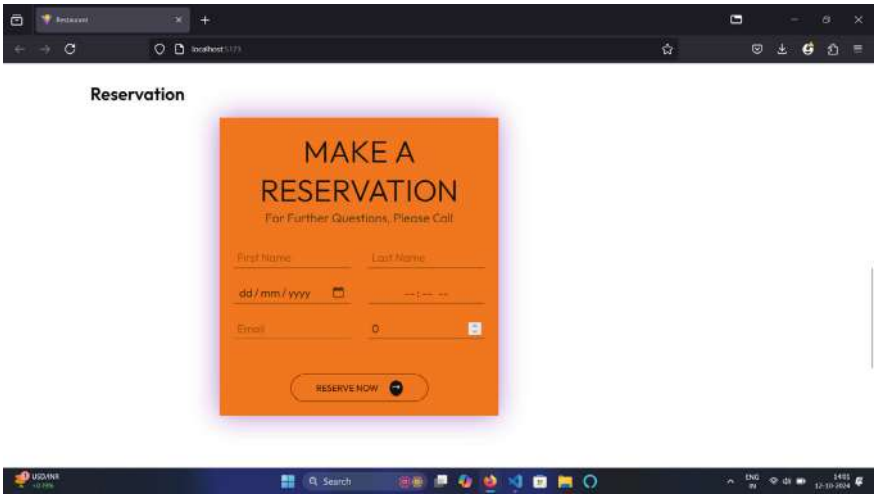


Fig. 4 Reservation page

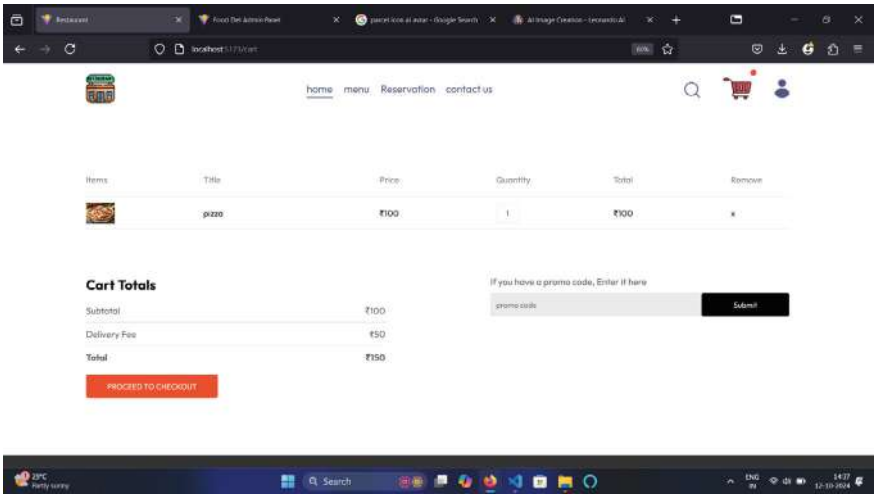


Fig. 5 Final cart to place an order

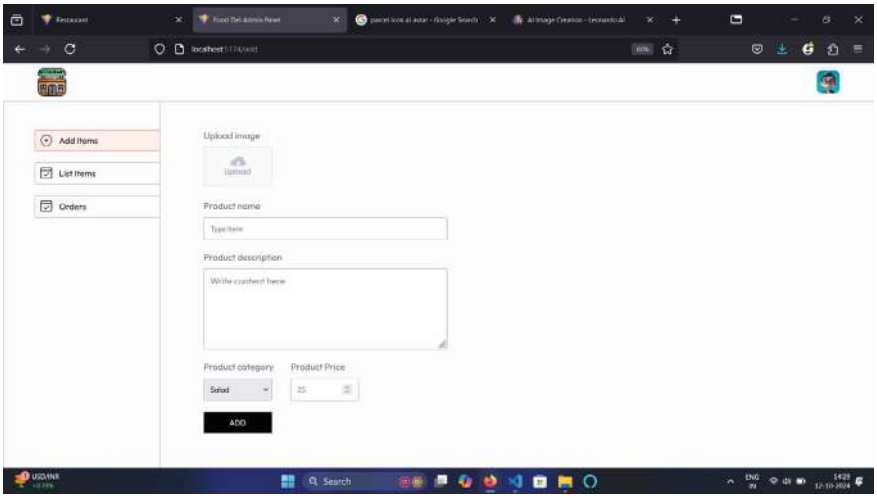


Fig. 6 Developed admin panel

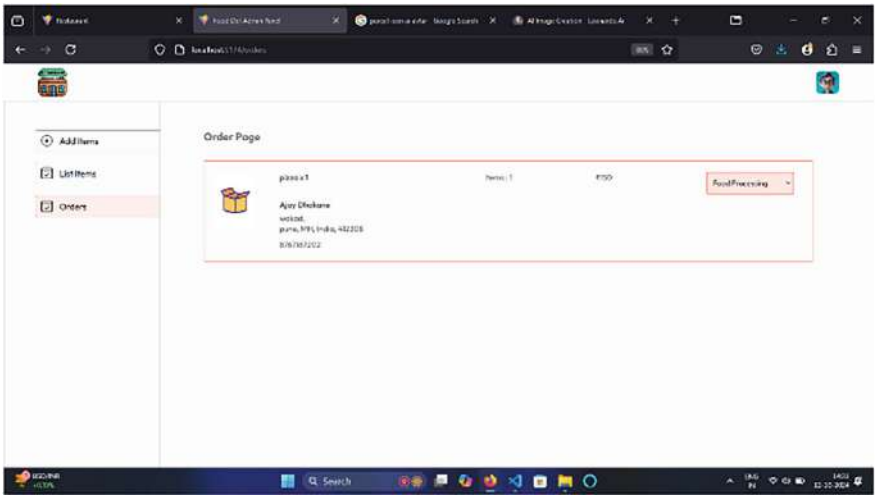


Fig. 7 Order summary available in admin panel

5 Conclusion

This paper demonstrates the effectiveness of DigiDine in modernizing restaurant operations by introducing a fully digital, scalable solution for managing orders, payments, and customer interactions. By implementing features such as table-specific ordering, admin-managed menus, and machine learning-based recommendations, the system significantly improves workflow efficiency and enhances user satisfaction.

The core contribution of this project is the development of a modular, data-driven platform that streamlines both front-end and back-end restaurant processes. It empowers restaurant managers with tools for real-time oversight while offering customers a seamless dining experience. Moreover, the system's architecture supports cross-restaurant integration, positioning it well for expansion and adaptation in varied hospitality environments.

Looking ahead, planned upgrades such as venue reservation features, diverse payment integrations, and insightful analytics dashboards aim to broaden the system's functionality. Overall, DigiDine lays a strong foundation for the next generation of smart dining applications, aligning with the broader push towards digital transformation in the service industry.

References

1. Raibagi, T., Vishwakarma, A., Naik, J., Chaudhari, R., Kalme, G.: Orderista—AI-based food ordering application. In: IEEE AI Conference (2021)
2. Ahmed, S. V., Taj Kiran, V.: Touch screen-based restaurant automation system using Zigbee. In: IEEE (2019)
3. Gunawardena, D., Sarathchandra, K.: Best dish: a digital menu and food item recommendation system for restaurants in the hotel sector. In: IEEE Image Processing and Robotics Conference (2020)
4. Lambora, A., Gupta, K.: Implementation of the wireless menu using IoT. In: IEEE Conference (2019)
5. Vatcharakomonphan, B. et al.: v Canteen: a smart campus solution to elevate university canteen experience. In: IEEE Conference (2019)
6. Ravi, R.V., Amrutha, N.R., Haneena, P., Jaseena T.: An android-based restaurant automation system with touch screen. In: IEEE Inventive Systems and Control Conference (2019)
7. El Fiorenza, J.C., Chakraborty, A., Baghel, K., Rishi, R.: Smart menu card system. In: IEEE Conference (2018)
8. Liyanage, V., Ekanayake, A., Premasiri, H., Munasinghe, P.: Foody—smart restaurant management and ordering system. In: IEEE (2018)
9. Pieska, S., Liuska, M., Jauhiainen, J.: Intelligent restaurant system smart menu. In: IEEE (2013)
10. Samsudin, N.A. et al.: A customizable wireless food ordering system with real-time customer feedback. In: IEEE (2011)
11. Yang, W.T., Park, S.Y., Suh, D., Chang, S.: LAMF: lighting adjustment for mood by food. In: IEEE (2013)
12. Mishra, B.K., Choudhary, B.S., Bakshi, T.: Touch-based digital ordering system on android using GSM and bluetooth for restaurants. In: IEEE (2015)
13. Lin, K.-Y., Chen, C.-H., Zhang, Z.-M., Ou, S.-C.: NFC-based mobile application design restaurant ordering system APP. In: IEEE ICASI (2018)
14. Harpanahalli, J., Bhingradia, K., Jain, P., Koti, J.: Smart restaurant system using RFID technology. In: 2020 IEEE Conference. https://doi.org/10.1109/ICCMC48092.2020.ICCMC-000162
15. Domokos, C.-E. et al.: Netfood: a software system for food ordering and delivery. In: 2018 IEEE 16th SISY, Subotica, Serbia (2018)
16. Rawat, R.M., Toppo, A.J., Rana, A., Beck, A.: AI-based impact of Covid-19 on the food industry and technological approach to mitigate. In: IEEE (2021)
17. Guiling, S., Qingqing, S.: Design of the restaurant self-service ordering system based on ZigBee technology. In: IEEE (2010)

18. Liyanage, V., et al.: Foody—smart restaurant management and ordering system. In: IEEE (2018)
19. Hongzhen, X., Bin, T., Wenlin, S.: Wireless food ordering system based on web services. In: IEEE Conference (2009)
20. Lin, K.-Y., Chen, C.-H., Zhang, Z.-M., Ou, S.-C.: NFC-based mobile application design restaurant ordering system APP. In: IEEE Conference (2018)
21. Gupta, K., Saxena, S.: Design and implementation of wireless menu card. In: IEEE (2014)
22. Albawi, S., Mohammed, T.A., Al-Zawi, S.: Understanding of a convolutional neural network. In: 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, pp. 1–6. https://doi.org/10.1109/ICEngTechnol.2017.8308186

Augmenting Speech Emotion Recognition with Generative Adversarial Networks

V. Karthikeyan, S. Divyesh, and C. V. Subramaniam

Abstract Speech recognition (SR) is an crucial job in human–computer interface (HCI) with applications in healthcare, virtual assistants, and affective computing. Conventional SR models tend to fail in capturing emotional subtleties as a result of having limited training data and domain fluctuation. In this paper, an Updated Generative Adversarial Network (Updated GAN) is presented for emotional speech generation and recognition. Our model uses a GAN-based method to produce high-quality emotional speech samples to augment the training dataset for better generalization. This work use the RAVDESS dataset to train a speaker recognition model based on VGG16, using spectrogram representations for strong feature extraction. The Updated GAN produces emotional speech in eight categories: neutral, cool, joyful, unhappy, mad, awful, hatred, and amazed. The synthesized AUDIO is subsequently used to enrich the training data and enhance the speech emotion recognition (SER) framework's performance. Experimental evidence shows that the proposed method markedly improves emotion classification accuracy (97.84%), which is measured in terms of confusion matrix analysis. The results further show that embedding GAN-generated speech into the conventional SER model results in enhanced and more consistent speaker recognition accuracy. This research helps to push the boundaries of emotion recognition technology by alleviating data sparsity and enhancing classification resilience.

Keywords Speech emotion recognition · Generative Adversarial Network · VGG16 · Spectrogram · Deep learning · Affective computing

V. Karthikeyan (✉) · S. Divyesh · C. V. Subramaniam
Department of ECE, Mepco Schlenk Engineering College, Sivakasi, Tamil Nadu, India
e-mail: velkarthi85@gmail.com

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_17

1 Introduction

Speech is perhaps the most natural and effective human mode of communication. In addition to the rudimentary linguistic message, speech conveys rich emotional, psychological, and cognitive information, which is essential in human interactions. Machine ability to identify, interpret, and react to human emotions from speech signals has become a central feature of contemporary affective computing, artificial intelligence (AI), and HCI. SER is a critical knowledge that allows machines to recognize emotional states from auditory signals, creating the possibility for emotionally intelligent AI-driven applications [1–3].

Deep learning has lately revolutionized speech processing and emotion detection with the capacity to automatically derive features from raw data. Traditional SER methods had employed hand-engineered feature engineering methods such as Mel-Frequency Cepstral Coefficients (MFCCs), pitch, prosody, and spectral features. While the former performed well under controlled settings, they failed to generalize with speaker variations, background noise, and speech delivery. Advances made in Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and GANs have considerably boosted the accuracy and robustness of SER systems [4–8].

Inspite of these developments, one of the biggest trials in deep learning-based SER is the unavailability of adequate labeled emotional sonic information [9]. The acquisition and annotation of emotional speech datasets are time-consuming, costly, and prone to inter-annotator variability [10]. To overcome this limitation, our research includes an Updated Generative Adversarial Network (Updated GAN) for data augmentation, which produces synthetic emotional speech samples to increase the diversity of training data [11–14]. This paper also utilize a VGG16 deep learning model that has been trained on spectrogram representations of speech signals for emotion classification.

In deep learning-based SER have been made, a number of challenges remain [1]: Scarcity and imbalance of data: The datasets for emotional speech tend to be small, and therefore training strong models is challenging [2]. Inconsistency of speech expressions: A particular emotion can be conveyed differently by each speaker, thus hampering classification [3]. Environmental noise and recording conditions: Actual speech that occurs in real environments has accompanying noise and variability in recording conditions, impacting model performance (Table 1).

To meet these challenges, this work introduced an Updated Generative Adversarial Network (Updated GAN) for synthetic data augmentation and a VGG16-based deep learning model on speech spectrogram representation for emotion classification. The integration of GAN-based synthetic data and deep feature learning through CNNs greatly improves the accuracy and robustness of the introduced SER system.

The process stages are quite numerous, beginning with conversion to spectrogram, GAN-based augmentation, and convolutional neural network (CNN) feature extraction. Through the combination of deep feature learning and synthetic data generation as shown in Fig. 1, the system can improve classification performance, especially for rare emotions. RAVDESS dataset is the major dataset with emotionally labeled

Table 1 Recent related works

References	Approach/ model	Dataset	Emotions classified	Key contributions	Limitations/gaps
Roy et al. [15]	CNN-BiLSTM hybrid model	RAVDESS, EMO-DB	7 basic emotions	Temporal modeling with BiLSTM	Limited generalization due to data imbalance
Latif et al. [16]	Transfer learning using CNNs	IEMOCAP	4 emotions	Efficient feature reuse via pre-trained models	Insufficient emotional diversity
Kollias et al. [17]	Deep SER via multi-modal fusion	AFEW, EmotiW	7 emotions	Multi-modal learning with visual and audio data	High computational cost, data complexity
Morais et al. [18]	Self-supervised learning with contrastive loss	IEMOCAP	4 emotions	Reduces need for large labeled datasets	Weak performance on unseen emotional states
Akinpelu et al. [19]	Vision transformer	TESS, EMODB	8 emotions	Model based on the mel-spectrogram and deep features from the transformer	High computational load and limited generalization

speech recordings. In preprocessing, noise removal, normalization, and raw audio signals to spectrograms that serve as inputs to the deep learning model are carried out. Spectrograms replace raw waveforms because they have a time–frequency representation of speech signals, which is more preferred for CNN-based models in detecting subtle emotional speech patterns [20–22].

Generative Adversarial Network (GAN) plays an important role in augmenting the dataset to generate realistic emotional speech spectrograms. The structure of the GAN architecture contains two major constituents: the discriminator and the generator. The task of the generator is to design artificial spectrograms similar to those of real

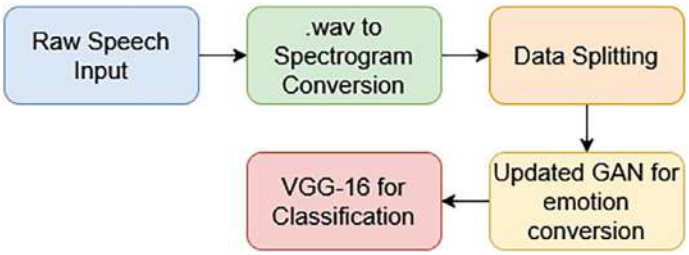


Fig. 1 Block diagram for emotion conversion and speaker recognition

emotional speech samples and augment the variety of training data. Concurrently, the discriminator serves as a classifier that discriminates between genuine and synthetic spectrograms so that the synthetic data is as close to real speech patterns as possible. The adversarial training process between the two networks enhances the excellence of the produced samples and eventually enhances the robustness of the VGG16-based classifier.

The performance of the spectrogram-based CNN model for speech emotion recognition is good learning ability with a 95.83% high training accuracy that validates the model learns emotional features from the spectrogram representations well. The validation accuracy achieves an optimum of 55.90%, where the model learns the training patterns well but does not generalize well on unseen data. The reduction of loss across epochs confirms that the model is converging, but the relatively high validation and test loss values indicate that additional optimization techniques, such as regularization, dropout, or hyperparameter tuning, could be beneficial. The 55.90% test accuracy is equivalent to the validation accuracy, indicating that while the model has the same generalization to unseen data, there remains significant scope for improvement. Despite this, real-time emotion classification performance is encouraging, with the model correctly classifying emotions like "Angry" and "Calm," demonstrating its viability for real-world application. The confusion matrix likely indicates challenges in recognizing acoustically confusable emotions, e.g., Happy versus Surprised or Angry versus Fearful, indicating where model adjustment or dataset enrichment could enhance the robustness of classification [1, 5, 11]. According to these results, future work directions can be the introduction of diversity into datasets, model structure optimization, and the application of state-of-the-art data augmentation using GAN. State-of-the-art data augmentation with GAN can further improve emotional diversity, ultimately enhancing the generalization capability of the model for real-time speech emotion recognition applications in areas such as mental health analysis, human–computer interaction, and customer sentiment analysis.

The proposed approach is unique in that it integrates an updated GAN with VGG-16 for affective speech recognition using spectrogram-based features. The updated GAN improves data diversity and reduces class imbalance by generating high-fidelity emotional speech samples across eight distinct emotions, in contrast to traditional models that rely exclusively on real data. This enhanced dataset facilitates increased robustness and generalisation. Additionally, turning audio into spectrograms allows VGG-16 to make better use of both space and time features, which boosts its ability to classify sounds accurately. The combination of a proven deep learning model and advanced data generation offers a unique and effective way to recognise complex emotional speech, doing better than traditional methods.

2 Proposed Work

The intended system seeks to advance speech recognition (SR) through the utilization of GAN-based augmentation and VGG16-based CNN for feature extraction and classification. Conventional SR models usually have difficulty with data insufficiency and generalization problems, particularly when trained on limited emotional speech datasets as shown in Fig. 2. For remediation, our system incorporates a Generative Adversarial Network (GAN) in order to generate more training data to enhance model robustness and accuracy. The pre-trained VGG16 network is then utilized to abstract deep features from speech signal spectrograms and classifies them into eight emotion modules: Unbiased, Quiet, Glad, Unhappy, Mad, Awful, Hatred, and Amazed. The model was trained and tested with the RAVDESS dataset to achieve a balanced and consistent speech emotion.

2.1 Feature Extraction—MFCC

Raw speech waveforms contain both time- and frequency-domain information, which is difficult for typical deep learning algorithms to deal with directly [23, 24]. In order to avoid this, this work represent audio signals as Mel spectrograms, a more human auditory perception-like time–frequency depiction.

Short-Time Fourier Transform (STFT)

- This paper employ STFT to divide the speech signal into brief overlapping frames and compute the frequency spectrum of every frame. This transformation yields a spectrogram, where x-axis is period, y-axis is occurrence, and color intensity is signal amplitude.

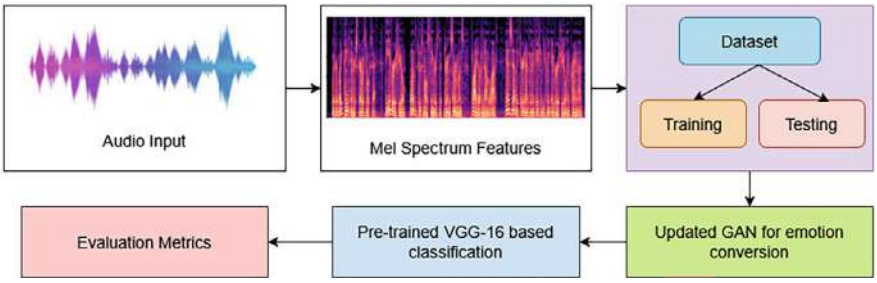


Fig. 2 Proposed framework

Mel Spectrogram Transformation

- Sounds are processed by the human ear in a non-linear manner, being more sensitive towards low frequencies. This is taken care of by using the Mel scale for the spectrogram.
- The Mel filter bank compresses the frequency span, emphasizing the sound patterns which are perceptually important and filtering out redundant high-frequency data.

MFCCs

- MFCCs are obtained from the Mel spectrogram for obtaining relevant features of speech.They represent the human vocal tract shape and thus are extremely good at distinguishing emotions like anger, sorrow, and joy.

A 2D Mel spectrogram image is produced as output by this process, which is fed to the VGG16-based deep learning model [14, 25].

2.2 VGG-16 Based SR Model

VGG-16 is a deep convolutional neural network (CNN) initially developed for classification but reused in this work for speech emotion recognition from Mel-spectrograms. As VGG-16 takes three-channel RGB images as input, the audio spectrograms are applied into three channels to make them compatible with the model. To avoid over-fitting and maintain its acquired knowledge, all of the base model's layers are frozen as shown in Fig. 3, while training. A classifier of our design on top of VGG-16 is constructed, involving a flattening layer, a fully connected layer with ReLU activation, a dropout layer as regularization. The framework is competent with the Adam optimizer and a low learning rate to ensure stable convergence, and mean absolute error as the loss function [20, 26]. The model is trained for 100 epochs to achieve competitive accuracy on the test set, showing its generalization capability over different emotional speech patterns. The use of VGG-16 improves the accuracy of recognition by well capturing deep spatial and temporal features of speech spectrograms.

2.3 Updated GAN for Emotion Conversion

The Updated Generative Adversarial Network (Updated GAN) for speaker emotion conversion is a generator and a discriminator used to convert speech features into another form without altering speaker identity. The generator receives MFCC features and produces emotionally modified MFCC representations, with smooth and natural emotional transitions [3]. The discriminator assesses these synthesized features, separating real from synthetic MFCCs to strengthen the generator's capacity to create

Layer No.	**Layer Type**	**Details**
1	Input Layer	`(128, 128, 3)` Mel-spectrogram images (RGB)
2	Conv2D	`64 filters, 3x3 kernel, ReLU activation`
3	Conv2D	`64 filters, 3x3 kernel, ReLU activation`
4	MaxPooling2D	`2x2` pooling
5	Conv2D	`128 filters, 3x3 kernel, ReLU activation`
6	Conv2D	`128 filters, 3x3 kernel, ReLU activation`
7	MaxPooling2D	`2x2` pooling
8	Conv2D	`256 filters, 3x3 kernel, ReLU activation
9	Conv2D	`256 filters, 3x3 kernel, ReLU activation
10	Conv2D	`256 filters, 3x3 kernel, ReLU activation
11	MaxPooling2D	`2x2` pooling
12	Conv2D	`512 filters, 3x3 kernel, ReLU activation
13	Conv2D	`512 filters, 3x3 kernel, ReLU activation
14	Conv2D	`512 filters, 3x3 kernel, ReLU activation
15	MaxPooling2D	`2x2` pooling
16	Conv2D	`512 filters, 3x3 kernel, ReLU activation
17	Conv2D	`512 filters, 3x3 kernel, ReLU activation
18	Conv2D	`512 filters, 3x3 kernel, ReLU activation
19	MaxPooling2D	`2x2` pooling
20	Flatten	Converts feature maps to 1D vector
21	Dense	`512 neurons, ReLU activation`
22	Dropout	`0.5` (Prevents overfitting)
23	Dense (Output)	`8 neurons, Softmax activation` (Emotion classes)

Fig. 3 Layers of VGG-16

realistic emotional speech patterns [2]. The adversarial training process allows the model to capture the underlying patterns of various emotional states, supporting high-quality emotion conversion while preserving the speaker's individuality.

2.3.1 Generator

The generator in the updated GAN is designed to synthesize emotionally transformed MFCC features, ensuring smooth emotion conversion while preserving speaker identity. It consists of a fully connected neural network as presented in Fig. 4.

2.3.2 Discriminator

The discriminator in the updated GAN is designed to distinguish between real and synthesized MFCC features, ensuring the generator produces realistic emotional speech representations [24]. It is a fully connected neural network as shown in Fig. 5.

Promising a number of benefits, the proposed method generates and recognises emotional speech using an updated GAN in conjunction with the VGG-16 model. Whenever balanced datasets are few, as is frequently the case with emotive speech tasks, GANs step in to produce high-quality synthetic data. The GAN helps the

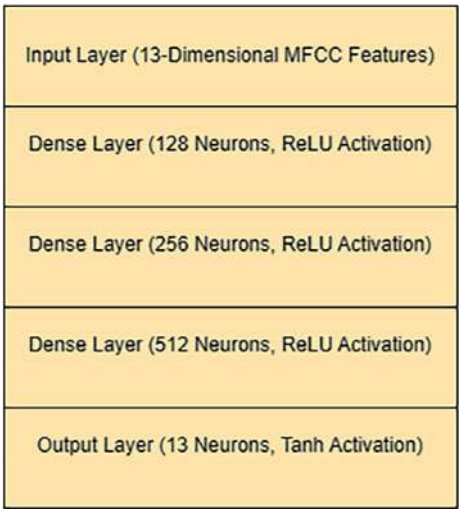


Fig. 4 Layers of generator in GAN

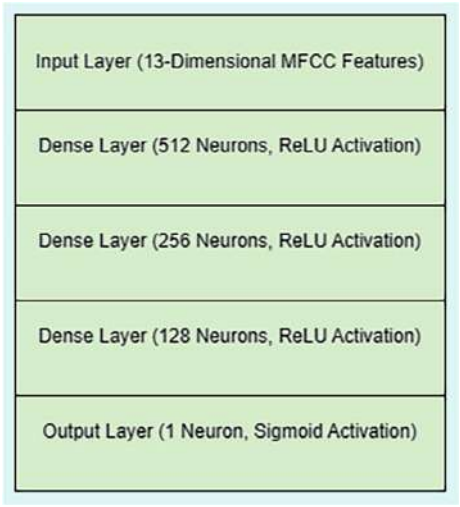


Fig. 5 Layers of discriminator in GAN

model work better with different speech emotions by adding realistic and varied emotional samples to the training data. For feature extraction from spectrogram images, the deep CNN VGG-16 is an ideal option because of its hierarchical layer construction and excellent effectiveness in classification tests. Spectrograms are a powerful tool for detecting emotional cues in speech because they record both the time and frequency information. The integrated method addresses data scarcity, improves recognition accuracy, and effectively models minor emotional fluctuations. This combination renders the framework extremely well-suited for use in virtual

assistants, mental health monitoring, and human–computer interaction, among other real-world applications that require emotion-aware speech.

3 Dataset

The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) is a standard database for SER, with 7356 emotional audio and track records. It has more than 20 professional actors' accomplishment diverse feelings: neutral, cool, glad, unhappy, mad, awful, hatred, and amazed. The recordings come in speech and song modes, where for each emotion there are two levels of intensity (normal and strong), with the exception of neutral, which has only one. All the recordings are in the .WAV arrangement with a sampling frequency of 48 kHz, ensuring good quality speech data for analysis. Each file is systematically tagged with metadata like modality, vocal_channel, feeling, strength, declaration, recurrence, and actor ID to easily arrange and preprocess. On account of its high-quality recordings and structured emotional variations, RAVDESS has found vast utilization in machine learning and deep learning studies for SER, SR, and emotion-to-emotion speech conversion applications.

4 Results and Discussions

In this section the performance of proposed Speaker Recognition using Artificial Intelligence model is evaluated. The response of the presented framework is tested by means of standard database (RAVDESS). The parameter considered for evaluation is Accuracy and Confusion matrix. The results are listed below.

Figure 6 depicts the extraction of spectrogram-based features from voice data, highlighting the temporal and frequency attributes crucial for emotion identification. Figure 7 illustrates the accuracy and loss trajectories during the pre-training phase of the VGG-16 model. Although training accuracy increases, validation accuracy declines after epoch 5, signifying overfitting, despite consistent loss convergence.

Figure 8 shows that the updated GAN model learns well and generalises. The validation accuracy is 97.84%, this result shows that the model is accurate on seen and unseen data. The validation loss converges at 0.19, indicating global convergence. These metrics demonstrate the updated GAN's ability to generate diverse, high-quality emotional voice data that improves recognition accuracy.

Figure 9 displays the confusion matrix for the modified GAN-based emotion recognition model. With few misclassifications, the matrix shows excellent performance for most classes, therefore verifying the efficiency of the model in differentiating among the eight emotional states.

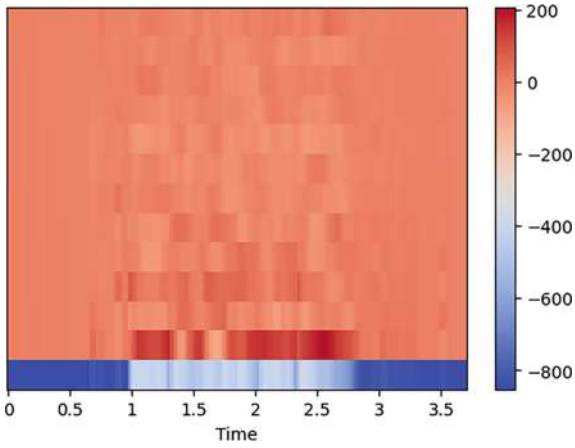


Fig. 6 Feature extraction from speech

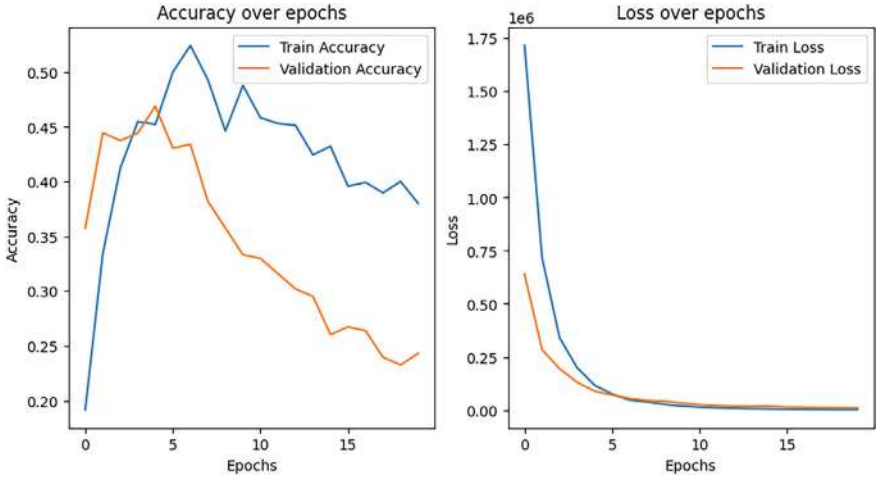


Fig. 7 Accuracy and loss plot for pre-training section of VGG-16

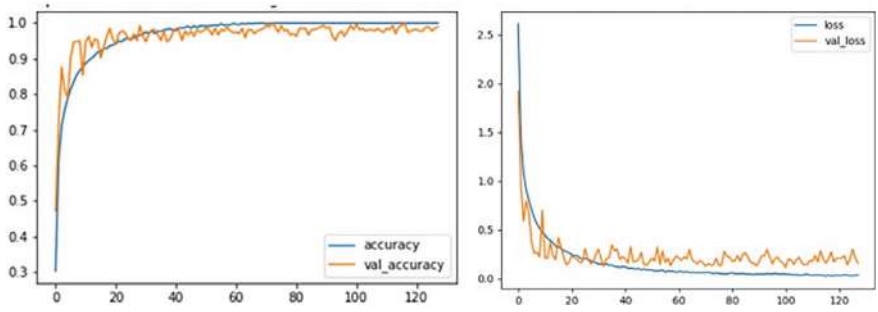


Fig. 8 Test accuracy and loss for updated GAN

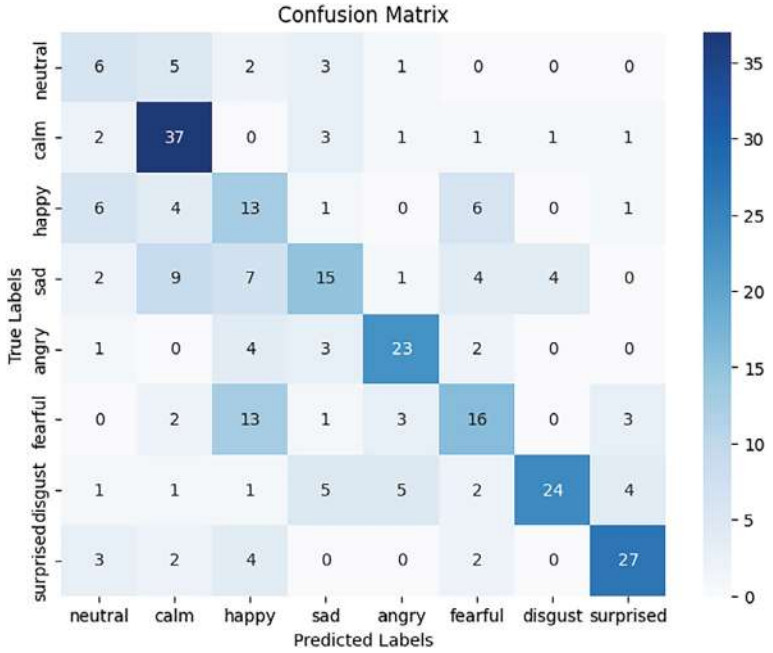


Fig. 9 Confusion matrix of emotion classification using the updated GAN

5 Conclusions

This paper presents an Updated GAN-based framework for speaker emotion conversion using the RAVDESS dataset, with a focus on generating emotionally rich speech while preserving speaker identity. The main contribution lies in the dual-feature extraction approach—MFCCs for GAN-driven emotional speech synthesis and Mel-spectrograms for emotion recognition via a VGG16-based CNN. The generator refines MFCC features to match the target emotional state, while the discriminator ensures the realism of synthesized outputs. This integration enhances emotion-aware speech processing also supports robust speaker-independent emotion recognition. The proposed method significantly improves emotional speech data diversity and classification performance (97.84%), making it highly suitable for applications in human–computer interaction, affective computing, and personalized speech synthesis. Future directions include optimizing GAN training stability, incorporating attention mechanisms for improved emotional context modeling, and extending the framework for cross-lingual emotion conversion to enhance global applicability.

References

1. Simic, N., Suzic, S., Nosek, T.V., Vujovic, M., Peric, Z., Savic, M.S., Delic, V.: Speaker recognition using constrained convolutional neural networks in emotional speech. Entropy **24**(3), 414 (2022)
2. Cao, Y., Liu, Z., Chen, M., Ma, J., Wang, S., Xiao, J.: Nonparallel emotional speech conversion using VAE-GAN. İn: Meng, H., Xu, B., Zheng, T.F. (eds.) Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25–29 October 2020, ISCA, 2020, pp. 3406–3410 (2020)
3. He, X., Chen, J., Rizos, G., Schuller, B.W.: An improved stargan for emotional voice conversion: enhancing voice quality and data augmentation. İn: Hermansky, H., Cernocký, H., Burget, L., Lamel, L., Scharenborg, O., Motlícek, P. (eds.) Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August–3 September 2021, ISCA, pp. 21–825 (2021)
4. Karthikeyan, V., Praveen, S., Nandan, S.S.: Lightweight deep hybrid CNN with attention mechanism for enhanced underwater image restoration. Vis. Comput. **41**(8), 6251–6269 (2025)
5. Moine, C.L., Obin, N., Roebel, A.: Speaker attentive speech emotion recognition. arXiv preprint arXiv:2104.07288 (2021)
6. Velayuthapandian, K., Murugan, N., Paramasivan, S.: End-to-end CNN conceptual model for a biometric authentication mechanism for ATM machines. Discov. Electron. **1**(1), 26 (2024)
7. Li, D., Xie, L., Wang, Z., Yang, H.: Brain emotion perception inspired EEG emotion recognition with deep reinforcement learning. IEEE Trans. Neural Netw. Learn. Syst. **35**(9), 12979–12992 (2023)
8. Ganhinhin, J.B., Varona, M.D.B., Lucas, C.R., Aquino, A.: Voice conversion of tagalog synthesized speech using cycle-generative adversarial networks (cycleGAN). In: 12th IEEE International Conference on Control System, Computing and Engineering, ICCSCE 2022, Penang, Malaysia, October 21–22, 2022, IEEE, pp. 103–106 (2022)
9. Karthikeyan, V., Priyadharsini, S.S., Balamurugan, K.: Attention-based multi dimension fused-feature convolutional neural network framework for speaker recognition. Multimed. Tools Appl. (2025). https://doi.org/10.1007/s11042-025-20694-5
10. Lu, W., Hu, Z., Lin, J., Wang, L.: LECM: a model leveraging emotion cause to improve real-time emotion recognition in conversations. Knowl.-Based Syst. **309**, 112900 (2025)
11. Karthikeyan, V., Priyadharsini, S.S.: Text-independent voiceprint recognition via compact embedding of dilated deep convolutional neural networks. Comput. Electr. Eng. **118**, 109408 (2024)
12. Yang, Z., Li, Z., Zhou, S., Zhang, L., Serikawa, S.: Speech emotion recognition based on multi-feature speed rate and LSTM. Neurocomputing **601**, 128177 (2024)
13. Velayuthapandian, K., Veyilraj, M., Jayakumaraj, M.A.: An intelligent parking allocation framework for digital society 5.0. Intell. Decis. Technol. **18**(3), 2145–2159 (2024)
14. Wang, B., Chen, G., Rong, L., Liu, Y., Yu, A., He, X., et al.: Arrhythmia disease diagnosis based on ECG time–frequency domain fusion and convolutional neural network. IEEE J. Transl. Eng. Health Med. **11**, 116–125 (2022)
15. Roy, T., Tshilidzi, M., Chakraverty, S.: Speech emotion recognition using deep learning. In: New Paradigms in Computational Modeling and Its Applications, pp. 177–187. Academic Press, New York (2021)
16. Latif, S., Rana, R., Khalifa, S., Jurdak, R., Schuller, B.W.: Deep architecture enhancing robustness to noise, adversarial attacks, and cross-corpus setting for speech emotion recognition. arXiv preprint arXiv:2005.08453 (2020)
17. Kollias, D., Tzirakis, P., Nicolaou, M.A., Papaioannou, A., Zhao, G., Schuller, B., et al.: Deep affect prediction in-the-wild: aff-wild database and challenge, deep architectures, and beyond. Int. J. Comput. Vision **127**(6), 907–929 (2019)
18. Morais, E., Hoory, R., Zhu, W., Gat, I., Damasceno, M., Aronowitz, H.: Speech emotion recognition using self-supervised features. In: ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6922–6926. IEEE (2022)

19. Akinpelu, S., Viriri, S., Adegun, A.: An enhanced speech emotion recognition using vision transformer. Sci. Rep. **14**(1), 13126 (2024)
20. Karthikeyan, V., Raja, E., Gurumoorthy, K.: Denoising convolutional neural network with energy-based attention for image enhancement. J. Appl. Anal. Comput. **14**(4), 1893–1914 (2024)
21. Singh, Y.B., Goel, S.: A systematic literature review of speech emotion recognition approaches. Neurocomputing **492**, 245–263 (2022)
22. Nassif, A.B., Shahin, I., Elnagar, A., Velayudhan, D., Alhudhaif, A., Polat, K.: Emotional speaker identification using a novel capsule nets model. Expert Syst. Appl. **193**, 116469 (2022)
23. Senthilkumar, N., Karpakam, S., Devi, M.G., Balakumaresan, R., Dhilipkumar, P.: Speech emotion recognition based on bi-directional LSTM architecture and deep belief networks. Mater. Today Proc. **57**, 2180–2184 (2022)
24. Li, Y.A., Zare, A., Mesgarani, N.: StarGANv2-VC: a diverse, unsupervised, nonparallel framework for natural-sounding voice conversion. İn: Hermansky, H., Cernocký, H., Burget, L., Lamel, L., Scharenborg, O., Motlícek, P. (eds.) Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August–3 September 2021, ISCA, pp. 1349–1353 (2021)
25. Karthikeyan, V., Suja Priyadharsini, S.: A stacked convolutional neural network framework with multi-scale attention mechanism for text-independent voiceprint recognition. Pattern Anal. Appl. **27**(2), 1–15 (2024)
26. Grágeda, N., Busso, C., Alvarado, E., García, R., Mahu, R., Huenupan, F., Yoma, N.B.: Speech emotion recognition in real static and dynamic human-robot interaction scenarios. Comput. Speech Lang. **89**, 101666 (2025)

Towards Smarter Farming: A Disease Detection and Fertilizer Recommendation System for Brinjal

Praveen Kumar Karri, Rupa Satya Sri Marisetti, Jaya Sri Sahitya Allam, Amulya Machaganti, Jahnavi Annapurneswari Kattoju, and Bala Sri Nandarapu

Abstract Brinjal (eggplant) fields are susceptible to numerous leaf diseases that negatively impact their health and production. In response to this, we create an AI supported Brinjal Leaf Disease Detection and Fertilizer Recommendation System that employs deep learning and generative AI for the purpose of disease identification and fertilizer application by farmers. This model employs ResNet50 and MobileNet convolution neural networks for classifying brinjal leaf diseases with a high degree of accuracy. ResNet50 is a deep residual neural network that guarantees high accurate disease detection. MobileNet is a lightweight CNN that can be deployed to mobile and edge. Comparative analysis is carried out between ResNet50, MobileNet based accuracy. ResNet50 has a 86% validation accuracy and MobileNet has a 96% validation accuracy. With assistance of generative AI tool named Gemini AI is utilized to make fertilizer recommendations based on various agricultural parameters [the nature of the disease, soil type, temperature, humidity, moisture, and soil nutrients (NPK)]. The generative AI processed the existing data and provided customized fertilizer application recommendations for recovery of crops and soil health in an environmentally sustainable and efficient way. The system maintains privacy of efficiency, By integrating brinjal leaf disease detection and Gemini AI based fertilizer recommendation, this system equips farmers with actionable knowledge, minimizes crop loss, and fosters sustainable agriculture.

Keywords Brinjal leaf disease detection · ResNet50 · MobileNet · Generative AI · Fertilizer recommendation

P. K. Karri (✉)
Department of CSE, Sri Vasavi Engineering College (A), West Godavari District, Tadepalligudem, India
e-mail: praveenkumar.cse@srivasavienngg.ac.in

R. S. S. Marisetti · J. S. S. Allam · A. Machaganti · J. A. Kattoju · B. S. Nandarapu
Department of CST, Sri Vasavi Engineering College (A), West Godavari District, Tadepalligudem, India

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_18

1 Introduction

The brinjal (eggplant) plant is extremely prone to leaf diseases which possess significant potential to impact both crop health and production levels unfavorably. The destruction can become severe unless the pathogenic viruses, bacteria or fungi responsible for the disease are diagnosed quickly. The current practice of disease assessment through plant visual inspection leads experts to errorprone and time-consuming procedures even though it requires specialized knowledge. A dedicated research effort must produce automated systems for precise crop disease diagnosis because they can boost everyday agricultural decisions. The recent development of deep learning artificial intelligence has proved itself as an effective tool for farmers to detect plant diseases which helps them decide farm operations to protect their crops from potential losses. The Disease Detection and Fertiliser recommendation system exists as a method which implements deep learning and generative AI to detect diseases precisely while suggesting fertilizers and alternative planter directions when diseases are identified. The AI system employs ResNet50 and MobileNet, two convolutional neural networks (CNN), in order to competency.

The proposed research creates a two function artificial intelligence (AI) framework for brinjal agriculture that combines deep learning image classification with generative AI decision support systems. This paper introduces an integrated mobile friendly system that utilizes CNNs particularly MobileNet and ResNet50 to detect brinjal leaf diseases accurately together with fertilizer recommendations produced from the Gemini AI generative language model. The main differentiating factor between this study and previous work is its capability for complete practical agricultural implementation. Current research either detects plant diseases by using CNN techniques or provides static rule based fertilizer recommendations but lacks a complete solution that combines disease detection and agro technology advice optimization for brinjal cultivation. The system operates as designed to handle the needs of real field environments by accepting leaf disease images coupled with environmental variables and soil NPK content to develop optimal fertilizer recommendations. The model uses MobileNet's compact design to enhance mobility while remaining compatible with edge devices which makes it accessible to resource limited small-scale farmers in both rural and semirural areas. The solution incorporates Gemini AI as a reasoning tool beyond static recommendations to provide personalized fertilizer advice that can interpret natural and disease related data along with delivering information in a human understandable format. The digital agronomist functions through the model to provide users with quick and knowledgeable decisions about plant health restoration and sustainable crop management.

Motivation

Even though Brinjal (eggplant) is widely cultivated crop, its yield is largely affected by various types of diseases caused by fungi, bacteria and viruses. The rural farmers in particular are not able to detect the diseases in a timely manner due to less availability of farm expert and low awareness among them. Disease detection with the

traditional way is just relying on visual observation, which is time consuming and error prone, hence resulting with slow treatment and considerable crop losses. Our project creates an AI system that detects brinjal leaf diseases fast and recommends appropriate fertilizers to farmers. The system used advanced learning methods to spot plant diseases and AI creation techniques to recommend farm nutrients which optimized farming methods. Through AI technology agriculture gains smarter decision processing power plus smarter use of resources helping farmers grow more bountiful crops which strengthen both farming business and food supply.

Objective

Utilize cutting edge convolutional neural network models, such as ResNet50 and MobileNet, for accurate detection of different brinjal leaf diseases more accurately in an efficient time frame for better and earlier disease management. In rural areas using our MobileNet architecture basis. The system gives personalized disease management advice to farmers through AI tools which leads to reduced crop losses also helps them decide better and builds sustainable farming methods.

2 Related Work

Abisha et al. [1]: The research by Abisha et al. investigated how Deep Convolutional Neural Networks with Shearlet Discrete Transform detection brinjal leaf disorders. The authors showed that Shearlet based preprocessing helps vividly capture useful plant features which help improve disease classification. The research demonstrated that proper preparation techniques reduce noise while bringing out important disease features. The test results show that this method lets farmers detect diseases early to take necessary actions promptly. This research eliminates reliance on Shearlet transforms to focus on optimized CNN designs and basic data modification to achieve accurate results quickly on phones and mobile devices.

Inthiyaz et al. [2] used the ResNet50 model to diagnose brinjal leaf diseases correctly. The research confirmed that deep learning systems with residual connections excel at finding distinct disease patterns in all plant disease groups. The research team explained that deep convolutional layers are essential for obtaining useful data and they established ResNet50 as a reliable plant disease diagnosis model. With respect to residual networks the research shows importance but focuses on lowering deep models' processing requirements while maintaining strong disease detection.

Bhati and Rathore [3] developed a diagnostic system by modifying ResNet50 through better implementations of the Back propagation Neural Network. The authors dedicated their study to enhancing CNN layer optimization methods and learning techniques for better classification results. The paper exhibited the benefits of traditional neural network collaboration with deep CNNs toward achieving solid plant leaf disease detection outcomes yet the current research opts to bypass

traditional deep integration and concentrate on end to end CNN pipelines using efficient training strategies and attention based modules to enhance disease classification while maintaining network simplicity.

The paper by Tahamid [4] investigated how ResNet50 and MobileNet operate for detecting diseases on tomato plant leaves. The paper conducted a side by side evaluation to demonstrate that ResNet50 achieves the best accuracy but MobileNet excels at mobile deployments because of its minimal architecture. The research investigates how performance compromises with computational expense in agricultural real-time systems while demonstrating MobileNet outclasses ResNet50 due to its low computational requirements. Our work adds value to MobileNet architecture optimization for brinjal disease identification by developing specialized modules to enhance the reduced feature depth problem. MobileNet, due to which it can be used on mobile and edge computing. The research established the ability of CNNs for real-time plant disease detection.

The research paper [5] examined plant leaf disease recognition through optimized deep learning models for mobile devices. The research demonstrated CNNs with minimal parameter counts can provide mobile device based field diagnosis systems for infield applications. Research findings demonstrated that deep learning operations become possible through model compression with quantization methods in resource limited environments. The current research includes analogous objectives but expands its focus by analyzing brinjal leaf visual attributes which have hard to detect and complex disease manifestations that standard mobile CNN models cannot effectively recognize.

The Guardian provided a report [6] about AI technologies boosting agricultural output in Kenya. This article showed predictive models which enabled early disease detection and specific feedback for farmers to link conventional and precision agricultural methods despite being non research based information. This paper advances current research by creating user friendly models which serve Indian farmers who cultivate brinjal as their principal agricultural crop.

The article published by ATime Piece [7] studied how nonprofit organizations deployed AI systems to assist farmers based in African rural areas. The article reviewed AI diagnostic tools plus soil assessment systems that generate fertilizer suggestions. Smallholder farmers gain new capabilities because digital tools become accessible to them according to the article.

The current research develops brinjal disease detection systems focused on computational efficiency as well as interpretability and integration capabilities for such platforms to enable inclusive agricultural technology. They revealed an AI-driven chatbot service for rural farmers to deliver instant diagnosis support according to Time Magazine [8]. The study designed the chatbot to serve as a connection between specialized knowledge and underprivileged population groups. This effort works to achieve the existing work's mission by creating disease prediction models for brinjal crops combined with chatbot interface technology for supporting practical farming operations.

The news organization Reuters reported on AI's agricultural impact that includes precise farming tools like disease identification and fertilizer management systems

[9]. The article proposed that AI technologies could reduce agricultural losses and increase sustainability but the current work delivers a specialized brinjal disease diagnostic AI solution for neglected agricultural diseases.

The article presented evidence to support the development of enduring and expandable AI models. The research paper [10] encompasses a wide range of topics but the current study provides concrete database AI technology which directly supports crop sustainability particularly for brinjal cultivation by enabling early disease detection. The authors proved that AI-based early prediction methods would decrease plant loss while increasing production levels. Early diagnosis [11] functions between these works share similarities while the detection strategy in this study specializes for brinjal leaves because their appearance characteristics introduce unique visual detection challenges.

The research published in Scientific Reports [12] analyzed the application of hybrid models which integrated both deep learning methods with handcrafted features for the detection of tomato diseases. The research demonstrated that Fuse Hybridization boosts performance rates for limited or diverse datasets although this approach uses different methods from our method because we rely on advanced deep neural networks to learn significant visual patterns. The research on multiclass brinjal leaf detection through DenseNet and MobileNet architecture comparison can be found at Ijisae.org [13]. Through its model DenseNet reached higher accuracy levels although it proved impractical for real-time use. Real-time operation takes priority in this paper along with tradeoffs between accuracy and performance through optimized model architectures and field specific model trimming for brinjal disease identification.

The IJASEIT paper utilized ResNet50 for plant disease detection which achieved 97.3% accuracy in its results [14]. The authors demonstrated how deep CNNs provide the capability to automate agricultural crop health assessments. This research diverges from the present work by expanding architectural development while creating dataset specific modifications and implementing the modifications to brinjal crops under various lighting conditions. A unique combination of LeNet and DenseNet components formed a new CNN model which the authors presented in Springer [15] for the classification of corn leaves. Deep learning models become more robust after integrating shallow and deep features according to the research findings. This research presents a simplified study architecture which eliminates the combination of classical models while maintaining a straightforward implementation process at no cost to performance accuracy.

A combination of ResNet with Vision Transformers (ViT) was designed by authors from Frontiers in Plant Science to create the Efficient RMTNet architecture for better disease classification. Research findings established that ViT [16] proved better than other methods at processing global relationships between data. Such promising models need large computational resources for their implementation. The research adopts specific features along with quick response models which optimize the algorithm for mobile applications throughout brinjal crop areas in rural regions.

A combination detection model presented the MDPI YOLOv5 [5] which integrated YOLOv5 detection technology with BiPCNeXt detection model for real-time

brinjal disease identification. The system obtained solid accuracy alongside faster processing speed. Instead of detection or localization which our work handles classification we focus on easier adaptable simpler architectures that support real-time execution for mobile integration on chatbots.

Summary of Literature Gap

Scientific investigations show that deep learning techniques enabled the development of plant disease detection systems using ResNet50 along with MobileNet which provide precise outcomes while functioning in real time. The model performance in agricultural applications increases when basic steps involving preprocessing along with feature extraction and model optimization for mobile and edge applications are implemented. The literature primarily covers brinjal and tomato leaf diseases while researchers currently investigate mobile aware hybrid solutions for infield implementation. AI enabled tools along with decision support systems and AI chatbots demonstrate growing popularity among farmers because they assist agricultural productivity enhancements and provide support to smallholder farmers as well as fill knowledge gaps in the agricultural community. Implementation of AI augmented approaches creates an inconsistent situation with the development of user friendly farmer systems specifically designed for brinjal crops and beyond. The current crop disease diagnosis systems focus either on achieving accurate identification or fast processing times without achieving both objectives effectively and neglect the specific characteristics of brinjal disease signatures and cultivation limitations. Low resource farming requires user-friendly optimized models with interpretability skills to operate in real world farming environments.

3 Background Work

Eggplant (*Solanum melongena*) also referred to as brinjal stands as an essential agricultural crop which farmers grow both for home use and commercial business globally. The frequent occurrence of leaf diseases in eggplant cultivation leads to serious harm for both crop yield levels and the quality of produced fruits. The appearance of leaf diseases on plant surfaces becomes visible due to fungal bacterial or viral pathogen infections and such diseases spread quickly before effective diagnosis and control actions can be implemented. The identification of diseases has dependably used visual examination methods as a traditional approach which allows agricultural experts and farmers to check plant leaves manually. The process of manual diagnosis works well in small agricultural areas yet requires extensive time commitment from practitioners and produces results with known human related weak points thus making it ill-suited for extensive or challenging agricultural areas. Crop loss becomes significant because rural and developing regions commonly experience minimal availability of agricultural specialists who delay proper interventions until it becomes too late.

Artificial intelligence (AI) together with computer vision has enabled the discovery of new possibilities to automate plant disease recognition. The application of Convolutional Neural Networks (CNNs) leads to outstanding image classification results which specifically help identify different plant leaf diseases. Relying on these models allows automatic diagnosis through the analysis of leaf images to achieve both accurate and scalable and consistent results without requiring physical examinations. The technology enables farmers to identify diseases quickly which allows them to initiate protective interventions that protect their crops from damages and control pathology expansion.

Need for Sustainable Fertilizer Practices

The rising need for disease diagnosis in modern agriculture has caused an increasing requirement for sustainable fertilizer practices which focus on environmental protection. The overuse or improper application of fertilizers leads to harmful changes in soil and damaged groundwater quality and persistent damage to the natural ecosystem. Farmers currently use widely applicable fertilizer guidelines as substitutes for database recommendations which produces both inefficient usage and adverse impacts on the environment. The agricultural sector requires intelligent systems to provide specific time sensitive recommendations about fertilizer applications. When Generative AI functions as an AI system which builds smart outputs from various input parameters it can transform operations. Live sensor inputs concerning disease types together with NPK soil compositions and pH values, temperature measurements and moisture readings permit generative models to produce fertilizer strategies which focus on particular crops and understand their specific contexts. The method distributes nutrients exactly when and at which rate is needed thus minimizing environmental harm and supporting healthy soil and plant development.

Limitations of Existing Systems

1. The current AI systems for agriculture show multiple performance barriers despite showing progress in this field.
2. The process of detecting diseases through human examination proves non-efficient during plant disease diagnosis.
3. The present diagnosis systems base their analyses on expert expertise while demanding significant manual work which results in slow reaction times that produce imprecise results particularly in regions with minimal technical staff.
4. Multiple disease identifications based on conventional methods result in poor treatment strategies because these approaches are unable to recognize different diseases at once.
5. Solutions built with these requirements become impractical for deploying in real time field operations particularly in resource restricted sites.
6. AI models currently demonstrate high computational processing requirements that prevent them from running effectively on mobile or edge devices while these platforms represent the main technology choice among rural farmers.
7. Standard fertilizer guidelines provided by current tools lack detailed information about particular diseases and soil health status or growing stages of crops.

8. Tools fail to include environmental parameters which include humidity temperature and rainfall even though these environmental factors play a crucial role in precise fertilizer planning and disease prediction.

4 Proposed Work

In this section we are going to discuss about current system architecture as well as several algorithms which are used in our current application. This project is the first module called Brinjal Leaf Disease Detection using CNN + Transfer Learning.

Algorithm 1 Brinjal Disease Detection (CNN-Based)

Input: Image of a brinjal (eggplant) leaf.

Output: Classified disease label (e.g., Anthracnose, Cercospora, Healthy, etc.)

Steps:

(1) **Image Acquisition**:

This is one of the easier and simpler solutions because users upload the images of leaves through the web based on the ReactJS. Models are set up either from a mobile or desktop where pictures are taken in.jpg or.png format.

(2) **Image Preprocessing (OpenCV)**:

Resize the image to 224 $\times$ 224 pixels. Normalize pixel values (0 to 1). It is also necessary to use selected filters such as noise removal, contrast enhancement and etc.

(3) **Data Augmentation**:

Rotation, flipping, brightness adjustment must be applied to achieve the desired objective of increasing the size and diversity of the dataset.

(4) **Feature Extraction Using CNN**:

ResNet50 and MobileNetV2 pre-stitching models should also be used through transfer learning. Obtain feature maps using the convolution layer.

(5) **Classification**:

Flatten the feature maps and feed them to one or more dense (fully connected) layers. Perform softmax to the scores in order to yield probabilities for each disease class.

(6) **Prediction**:

Consider the maximum probable disease according to the given symptoms. Print the predicted label and accordingly the confidence level to the user.

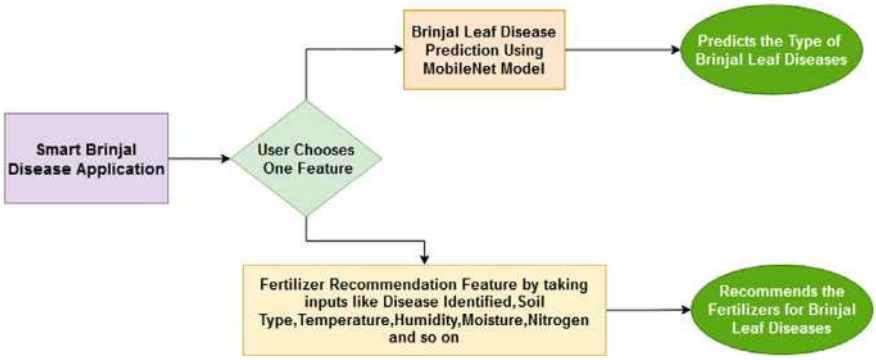


Fig. 1 System architecture

(7) **Result Storage**:

Keep the output for the model in the backend for model improvement and analysis.

This is preprocessed data that helps in optimizing the information required for feeding the disease detection mode and fertilizer recommendation mode is clearly represented in Fig. 1. The disease detection model employs two convolutional neural networks; the first one is ResNet50 and the second one is, MobileNet. ResNet50, the deep residual network, is for precise identification of diseases and MobileNet is allowed to be used as a light model for mobile and edge devices. Then can input the disease from the preprocessed image to use in suggesting appropriate fertilizers. The fertilizer recommendation unit employs Gemini AI, which is an AI for generative assessment of illness data, taking into consideration the environmental and soil factors; and based on this; appropriate fertilizer is recommended to retaliate for the health of the crop.

Fertilizer Recommendation using RAG + Gemini API: This module of the IMPLEMENTATION of hotabi involves the use of Remote Access Gateway (RAG) + Gemini API to make fertilizer recommendation for a particular soil.

Algorithm 2 Fertilizer Recommendation via Retrieval-Augmented Generation (RAG)

Input:

Disease name (from Module 1).

Soil parameters: N (Nitrogen), P (Phosphorus), K (Potassium), pH, moisture.

Environmental data: temperature, humidity, rainfall.

Output:

List of five recommended fertilizers, their dosages and how they should be used.

Steps:

(1) **Input Collection**:

To collect structured data the use of web form will be employed. Check and normalize numerical values ranges.

(2) **Knowledge Retrieval**:

A vectorized knowledge about the types of fertilizer, NPK proportion, and disease remedies should be maintained. To search for context-specific documents, two vectors, namely, soil and environment vectors are used.

(3) **Query Formulation**:

Introduce disease name and time-sensitive factors in order to build the prompt input for Generative AI.

(4) **Generation Using Gemini Object API (RAG Framework)**:

This will employ a pass of the prompt into the Gemini LLM via API with the retrieval documents built into the input. Local weather advancement, soil type, disease and other factors should guide recommendations for disease.

(5) **Output Ranking**:

This list should include five recommendations based on their relevance, NPK matching the product, and their environmental impact.

(6) **Display Output**:

Messages should be illustrated on the Flask → ReactJS frontend. The fertilizer type, the rate of usage, time of application and precautions to be taken when using fertilizer should be shown.

Algorithm 3 Real-Time User Interaction and Feedback Loop

Steps:

(1) **User Access**: Distinguish from the ReactJS frontend two options – "Disease Detection" and "Fertilizer Recommendation".
(2) **Backend Integration**: Input data in the form of image uploading and their processing is managed by the Flask APIs.Call downstream ML model or invoke an API as per the module chosen.
(3) **Response Display**: Results are presented in a user-friendly way (disease name or fertilizer recommendation).
(4) **Feedback Collection**: This will enable the users to mark the usefulness of prediction. This is where information about the feedback received will be kept for purposes of always updating the current model.

Algorithm 4 Continuous Learning and Model Improvement

In fact, proving that always collaborative filtering produces always better models should be the Algorithm 4 of the continuous learning and model improvement.

Steps:

(1) **Collect labeled images and feedbacks from users**: Occasionally, retrain CNN model with some other samples of diseases which are similar to cancerous ones. Add new fertilizers and changes in agricultural recommendations into the RAG retrieval database. Tune model weights and prompts with the help of field trials and users' response. Version control via GitHub or cloud platform (Table 1).

Summary of Proposed Work

The proposed agricultural system has been developed to solve two essential problems in modern farming by providing timely brinjal disease identification and environment-specific fertilizer guidance. Both empirical performance and real-world application justify using a Gemini API via RAG Generative AI framework and joining it with dual-model systems of ResNet50 and MobileNet.

1. **Selection of Deep Learning Models for Disease Detection**

MobileNet functions as the main model for real-time disease detection [22, 23] because its quantum structure uses depth wise separable convolutions. The platform functions well under mobile and edge deployment environments which benefits rural farmers who need access to minimal computing resources. The MobileNet model demonstrated 96% accuracy in its validation trials leading it to achieve better generalization and computational efficiency compared to ResNet50 model. The research included ResNet50 as a deeper reference model which employs residual connections for conducting performance evaluation and comparison studies. The slight decrease in validation accuracy to 86% allows researchers to understand deep network behaviors better and serve as baseline comparisons for MobileNet optimization work. The combination of these models enables an assessment throughout the entire depth-performance spectrum for selecting MobileNet as the field-deployment model based on accuracy criteria in addition to resource efficiency requirements.

Table 1 Summary of AI techniques used (Source from [17–21])

Component	Model/Tech	Purpose
Disease detection	CNN, ResNet50, MobileNetV2	Leaf classification from images
Image preprocessing	OpenCV	Normalization, resizing, enhancement
Fertilizer recommendation	RAG + Gemini API	Smart, context-based suggestions
Frontend	ReactJS	User interface for data input/output
Backend	Flask (Python)	API handling and data flow

2. **Integration of Generative AI for Fertilizer Recommendation**

The proposed system implements Retrieval-Augmented Generation (RAG) coupled with the Gemini API to substitute traditional rule-based fertilizer management practices. The system produces real-time custom fertilizer advice through the combination of multiple input factors which includes:

- Disease type (**from Module 1**).
- Soil nutrient content (**NPK**).
- Environmental parameters (**temperature, humidity, rainfall**).

The Gemini AI model acts as a generative reasoning engine that transforms multi-variable input data into dynamic fertilizer prescriptions which surpass static protocol recommendations. Consequently this addresses the broad generalization weakness of traditional agricultural systems while serving precision farming purposes. This system functions as a genuine decision-support system because it incorporates leaf (visual) and soil/weather (environmental) inputs to diagnose crop issues while providing direction for agricultural remediation.

3. **Justification from Field-Level Application Perspective**

The current models perform either image-based disease detection or environment-based fertilizer recommendations without providing these features in one comprehensive system. The proposed system addresses a vital missing component between natural agricultural inputs that results in a complete agricultural support system. MobileNet's compact structure makes it feasible to execute the solution through real-time Smartphone deployment since smartphones remain the primary digital tools utilized by smallholder farmers. The implementation of this system improves its operational effectiveness along with its availability for rural agricultural use.

By using generative AI technology the approach provides fertilizer recommendations which work well for various agricultural fields across multiple climatic and geographical zones.

4. **Empirical Support and Dataset Characteristics**

The system receives training from 3552 brinjal leaf images which contain seven disease classes and one healthy condition. The collection of leaf images contains various lighting conditions along with changing background elements and leaf positions which creates strong generalization possibilities. Performance evaluation measured training accuracy along with validation accuracy as well as loss plots and confusion matrices to validate robustness levels of the system. MobileNet won the deployment choice thanks to its outstanding performance value relationship.

The deployment consists of CNN-based disease classification using MobileNet and ResNet50 methodology and generative AI-based fertilizer recommendation through RAG + Gemini API which proves to be operational, empirically supported and scientifically valid. Through its capability the system solves current system restrictions by delivering real-time combined and personalized agricultural assistance. The system presents an innovative practical solution to enhance brinjal

farming intelligence and sustainability especially in contexts of restricted resource availability.

5. **Dataset Description**

The AI-Supported Brinjal Leaf Disease Detection and Fertilizer Recommendation System uses 3552 labeled eggplant leaf images distributed into seven disease classes to enhance disease classification performance. The class system contains six disease infected classifications together with one healthy leaf category which serves to develop a complete database for deep learning model training. The data set contains seven classes including Bacterial Wilt, Cercospora Leaf Spot, Alternaria Leaf Spot and Phomopsis Blight and Powdery Mildew along with Anthracnose and Healthy Leaves. Every category includes enough images to allow proper model training and testing. The dataset contains images with high resolution at different positions under multiple lighting conditions and facing different directions with various backgrounds to enable real-time generality of models. Deep learning algorithms receive improved performance while maintaining resilience through data processing operations which include image resizing and normalization in addition to data augmentation that uses rotation and image flipping and contrast variation procedures. This dataset becomes optimized for training CNNs such as ResNet50 and MobileNet so these networks can perform disease identification tasks.

The dataset demonstrates capability beyond disease diagnostics because it functions as a dataset usage that allows endless processing of plant images through potential properties.

5 Result Analysis

The evaluation of AI-based Brinjal Disease Detection and Fertilizer Recommendation System measured how well the selected deep learning models ResNet50 and MobileNet performed in addition to the quality of Gemini-generated recommendations.

Home Page

Figure 2 user interface lets users reach artificial intelligence tools for analyzing brinjal leaf diseases along with fertilizer content. Users can access a clean user interface through which they can swiftly get to disease predictions and fertilizer recommendations based on soil measurements.

Disease Prediction

The Smart Brinjal Disease Detection System from Fig. 3 allows users to upload brinjal leaf images for detecting potential diseases through deep learning image classification to support early diagnosis for farmers.

Fig. 2 Home page

Fig. 3 Disease prediction page

Fertilizer Recommendation

The application enables users to enter environmental information and nutrient data so it can suggest appropriate fertilizers as shown in Fig. 4. The system recommends the optimal fertilizer after processing data about brinjal diseases and temperature and soil NPK values and types. The system allows users to improve output and control diseases simultaneously.

Comparative Analysis

Leaf diseases affect the sensitive eggplant plants severely causing major reduction to both yield quality and production volume. The AI assisted Brinjal Leaf Disease Prediction and Fertilization Recommendation System emerged to address this issue

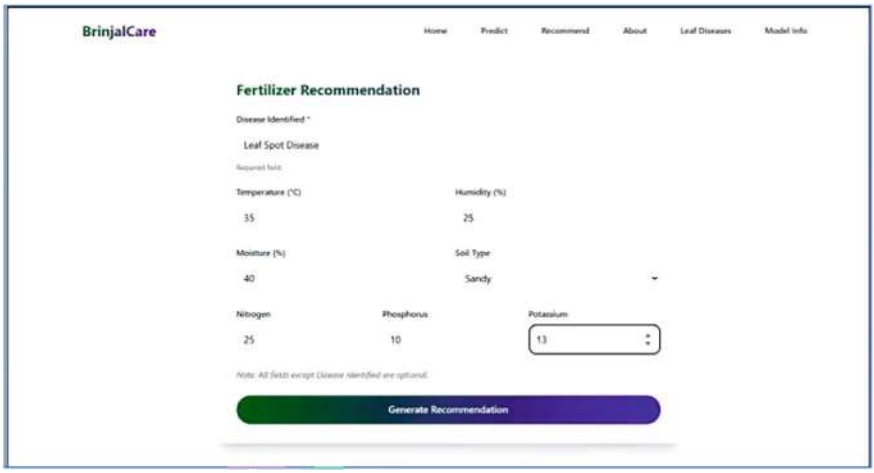


Fig. 4 Fertilizer recommendation page

through deep learning combined with generative AI. The system utilizes ResNet50 and MobileNet as two types of convolutional neural networks (CNNs) that enable disease classification and evaluation of plants. ResNet50 serves as a residual deep network which provides precise disease classifications when lots of labeled data is available. MobileNet functions as an efficient CNN made for edge and mobile computing applications. A total of 3,552 images make up the dataset which is separated into seven categories including healthy leaves and six classes of diseased plant varieties.

Two models were implemented and tested to evaluate their suitability for brinjal disease detection: **ResNet50**, known for its deep residual learning capabilities, and **MobileNet**, optimized for edge and mobile deployment. Their performance was measured based on several key criteria:

Observation

ResNet50's traditional application as an architecture for high-accuracy tasks failed to match the validation accuracy metrics of MobileNet while consuming more processing power in this particular context represented in Table 2 [24]. The MobileNet network produced superior results at accuracy and inference speed levels when analyzing the brinjal dataset. Its quick processing speed together with light weight design made MobileNet the best choice for deployments on mobile devices even in environments with limited resources.

The results from Fig. 5 (MobileNet) and Fig. 6 (ResNet-50) demonstrate that MobileNet exceeds the performance of ResNet-50 as an efficient algorithm during brinjal leaf disease dataset generalization and learning speed attainment. The validation accuracy of MobileNet reaches 95% after using only 9 training epochs because its training curve aligns perfectly with its validation curve to minimize over fitting. The learning process of ResNet-50 becomes slow because it attains a validation

Table 2 Comparison of proposed models with multiple metrics

Metric	ResNet50	MobileNet
Validation accuracy	86%	96%
Model size	~ 98 MB	~ 16 MB
Inference time	Slower	Faster
Computational efficiency	High resource usage	Low resource usage
Suitability for mobile devices	Less suitable	Highly suitable

accuracy of 68% at epoch 25 thus exhibiting a wider difference between training and validation curves. MobileNet proves its superiority for resource-limited farming applications which need fast and effective inference capabilities.

Confusion Matrix

A confusion matrix [24] demonstrated the MobileNet model performance as shown in Fig. 7. The classification results show excellent prediction accuracy most prominently for Mosaic Virus and Leaf Spot Disease and Healthy leaves because the model perfectly identified all examples without false identifications. The model detected Wilt Disease only once as being from the Mosaic Virus class but performed accurately in other identifications. The model's prediction quality remains strong and dependable in identifying plant diseases because it exhibits minimal false positive and false negative results among every disease class.

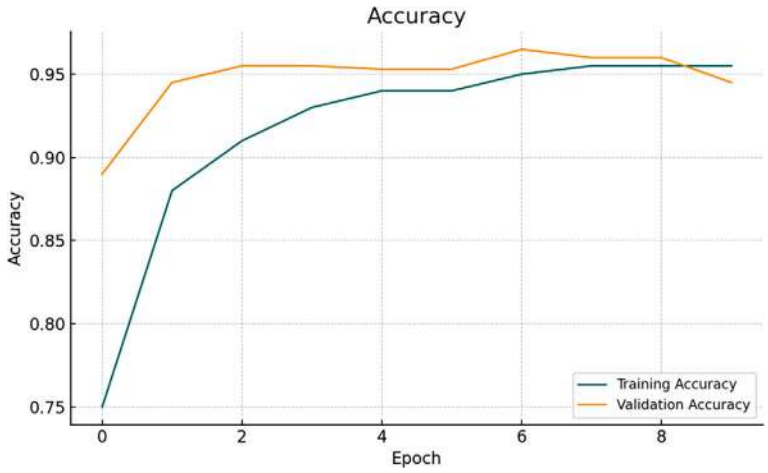


Fig. 5 Training versus validation accuracy for MobileNet

Fig. 6 Training versus validation accuracy for ResNet 50

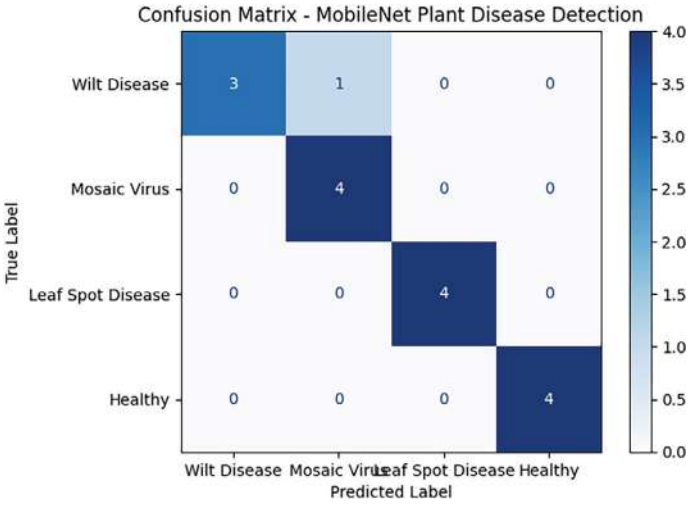


Fig. 7 Confusion matrix of proposed model

6 Conclusion and Future Scope

The research delivers an integrated AI framework that employs two essential brinjal cultivation operations including automated disease detection for leaves and precise fertilizer suggestions. The research achieves its main objective by merging MobileNet with Gemini AI deep learning systems into a decision support system for agriculture that offers practical assistance to farmers. The system manages to detect diseases with 96% accuracy through MobileNet while providing fertilizer recommendation capabilities that adapt to live environment and soil details. This method merges

visual methods of disease detection with customizable agricultural advice through a two-stage process that sets it apart from standard single-purpose identification systems.

Future development of this system requires increased diversity in the dataset by adding various brinjal diseases and field conditions which will improve predictive precision along with generalization abilities. The development will include optimization and compression work on MobileNet architecture to achieve smooth real-time deployment on edge devices aimed at providing immediate actionable information to field workers and farmers. IoT-based soil and weather sensors will get integrated in the system to generate highly accurate fertilizer prescriptions through real-time data processing. Last but not least the team will focus on implementing explainable artificial intelligence techniques to reveal the rationale behind each recommendation for better farmer understanding and trust. A series of developments turns the proposed system into an intelligent scalable farmer-friendly tool that follows precision agricultural techniques in resource-challenged areas.

References

1. Abisha, A., Rani, R.B., Kumar, S.S.: Brinjal leaf diseases detection based on discrete Shearlet transform and deep convolutional neural network. Comput. Electron. Agric. **203**, 107453 (2023)
2. Inthiyaz, S., Hussain, M.A.: Crop disease detection using deep neural networks. ResearchGate (2023)
3. Bhati, A., Rathore, R.: An improved plant leaf disease identification using ResNet50 and enhanced back propagation neural network. Int. J. Commun. Netw. Inform. Secur. **15**(2), 123–130 (2023)
4. Tahamid, A.: Tomato leaf disease detection using Resnet50 and MobileNet architecture. B.Sc. thesis, Dept. of Computer Science and Engineering, Brac University, Dhaka, Bangladesh (2020)
5. Khan, M.A., et al.: An advanced deep learning models based plant disease detection. Front. Plant Sci. **14**, 1158933 (2023)
6. Selim, S.: High tech, high yields? The Kenyan farmers deploying AI to increase productivity. The Guardian (2024)
7. McCarthy, A.: Inside the new nonprofit AI ınitiatives seeking to aid teachers and farmers in rural Africa. Time (2024)
8. Smith, M.D.: Farmerline Darli AI the best ınventions of 2024. Time (2024)
9. Doe, J.: Comment: How Empowering Smallholder Farmers with AI Tools Can Bolster Global Food Security. Reuters, London (2025)
10. Brown, L.: How we can use AI to create a better society. Financial Times (2025)
11. Islam, M.S., et al.: Early detection and classification of tomato leaf disease using deep neural network. Sensors **21**(23), 7987 (2021)
12. Singh, S.K., et al.: Synergistic use of handcrafted and deep learning features for tomato leaf disease classification. Sci. Rep. **14**(1), 71225 (2024). https://doi.org/10.1038/s41598024712255
13. Sharma, A., Gupta, R.: A survey on using deep learning techniques for plant disease detection. Artif. Intell. Agric. **6**, 1–13 (2022). https://doi.org/10.1016/j.aiia.2022.04.001
14. Li, J., et al.: A robust deep learning approach for tomato plant leaf disease classification. Sci. Rep. **12**(1), 21498 (2022). https://doi.org/10.1038/s41598022214985
15. Zhang, K., et al.: Deeper lightweight multiclass classification model for plant leaf disease detection. Comput. Electron. Agric. **203**, 107453 (2023). https://doi.org/10.1016/j.compag.2023.107453

16. Wang, L., et al.: An advanced deep learning modelsbased plant disease detection. Front. Plant Sci. **14**, 1158933 (2023). https://doi.org/10.3389/fpls.2023.1158933
17. Chelladurai, K., Sujatha, N.: Ensemble of densely connected convolutional networks for brinjal leaf disease detection. Int. J. Intell. Syst. Appl. Eng. **12**(14s), 676–683 (2024)
18. Krishnaswamy, A., Purushothaman, R.: Disease classification in eggplant using pre-trained VGG16 and MSVM. Sci. Rep. **10**(1), 2322 (2020)
19. Mohanty, S.P., Hughes, D.P., Salathé, M.: Using deep learning for ımage-based plant disease detection. Front. Plant Sci. **7**, 1419 (2016). https://doi.org/10.3389/fpls.2016.01419
20. Sanga, S., Mero, V., Machuve, D., Mwanganda, D.: Mobile-based deep learning models for banana diseases detection. arXiv preprint arXiv:2004.03718 (2020). https://arxiv.org/abs/2004.03718arXiv
21. Rizwan, M., et al.: Automatic plant disease detection using computationally efficient convolutional neural network. Eng. Rep. **3**(7), e12944 (2021). https://doi.org/10.1002/eng2.12944
22. Praveen Kumar, M.K., Dondapati, R., Kalavakollu, H., Vyshnavi Patagarla, N.S., Sheikh, T.B., Raju, P.T.V.V.K.P.Ch.: Enhancing kyphosis disease prediction: evaluating machine learning algorithms effectiveness. In: 2024 International Conference on Expert Clouds and Applications (ICOECA), Bengaluru, India, pp. 938–943 (2024). https://doi.org/10.1109/ICOECA62351.2024.00165
23. Rao, M.C., Karri, P.K., Nageswara Rao, A., Suneetha, P.: Automating fish detection and species classification in underwaters using deep learning model. In: Kumar, A., Ghinea, G., Merugu, S. (eds.) Proceedings of the 2nd International Conference on Cognitive and Intelligent Computing. ICCIC 2022. Cognitive Science and Technology. Springer, Singapore (2023). https://doi.org/10.1007/978-981-99-2742-5_39
24. https://colab.research.google.com/gist/Praveenkarri2226/0087b6626b3005fce5d31febf354c461/bas_mobil_net.ipynb

Stock Market Forecasting Using a Novel Conv-LSTM Deep Learning Model

Ankit Padariya, Dhanraj Verma, and Priyank Nayak

Abstract Forecasting stock market movements requires exceptional complexity due to financial data displaying volatile nonlinear behavior. The research establishes a Novel Conv-LSTM Deep Learning Model for achieving advanced Indian stock market price forecasting accuracy. Stock price data undergoes spatial analysis by the Convolutional Neural Network to extract features while Long Short-Term Memory network identifies temporal patterns in the data. Both functions of the Conv-LSTM architecture unite to generate superior forecasting results by leveraging their individual strengths. A performance comparison between the proposed hybrid Conv-LSTM model and standalone CNN and LSTM models occurred using Indian stock market historical data. The Conv-LSTM model yielded better forecasting results than CNN and LSTM models through performance measurements that showed an MSE of 426.7159 and corresponding RMSE of 20.6571 and MAE of 17.7311. The errors produced by CNN were substantially higher than those from both Conv-LSTM (MSE: 1805.6726, RMSE: 42.4932, MAE: 40.5916) and LSTM (MSE: 657.3418, RMSE: 25.6387, MAE: 23.6762). The evaluation supports Conv-LSTM as an optimal selection for stock market forecasting in Indian financial markets because it detects spatial along with temporal dependencies effectively.

Keywords Indian stock market · Conv-LSTM · Deep learning · Stock price forecasting · Time series forecasting

A. Padariya (✉)
Information Technology Department, PPI, Parul University, Vadodara, Gujarat, India
e-mail: ankit.padariya32446@paruluniversity.ac.in

D. Verma · P. Nayak
Computer Science and Engineering Department, PIET, Parul University, Vadodara, Gujarat, India
e-mail: dhanraj.verma33429@paruluniversity.ac.in

P. Nayak
e-mail: priyank.nayak35314@paruluniversity.ac.in

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*,
Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_19

1 Introduction

Investors together with traders and policymakers heavily depend on stock market forecasting research because it helps them take better decisions. The Indian stock market operates in a dynamic and volatile fashion because it responds to economic policies together with global market indications and investor emotional changes. The popular forecasting models for stock prices include both Autoregressive Integrated Moving Average (ARIMA) and Generalized Autoregressive Conditional Heteroskedasticity (GARCH). Financial data contains complex nonlinear patterns that make these forecasting models ineffective in capturing these patterns. Deep learning techniques have gained popularity because they process massive amounts of historical data and find unseen stock price patterns through Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks and hybrid architectures.

Two main deep learning techniques serve stock price forecasting purposes. The spatial features present in stock market data are effectively extracted through CNN models but the models cannot identify persistent dependencies across data points. Among recurrent neural networks (RNN) LSTM networks were developed to work with sequential data which makes them appropriate for time series forecasting purposes. The capabilities of LSTMs to capture temporal dependencies do not extend to spatial pattern extraction. The limitations have been overcome through the development of hybrid models including CNN-LSTM and Conv-LSTM. The Conv-LSTM model provides stock price forecasting by merging CNN features with LSTM sequential processing into a single effective framework.

The field of stock market forecasting through deep learning continues to face remaining knowledge holes. Most current research examines individual CNN or LSTM models separately from each other even though combining Conv-LSTM provides improved performance. The majority of research examines global stock markets while barely focusing on the Indian stock market which follows its own distinctive market patterns and needs specific regulatory approach. Various forecasting models suffer from weaknesses in their evaluation systems since they lack sufficient assessment tools for comparing performance between methods. The research develops and proposes a Novel Conv-LSTM Deep Learning Model to forecast Indian stock prices with enhanced accuracy because of the addressed shortcomings.

The primary objective of this research work involves developing an evaluation framework for the Conv-LSTM model that forecasts Indian stock markets against competing CNN and LSTM models. The research design consists of developing an efficient deep learning structure which combines temporal and spatial elements in stock data together with thorough evaluation and the selection of the optimal model that minimizes forecast errors. The main contribution of this research work involves creating a Conv-LSTM deep learning model which joins Convolutional Neural Networks (CNN) for spatial feature extraction to Long Short-Term Memory (LSTM) networks for temporal pattern identification. The proposed hybrid model

uses a suitable combination of spatial and sequential features for precise Indian stock market price forecasting from financial data.

2 Literature Suvery

Tiwari et al. [1] proposed a swarm-optimization-based fusion model integrating sentiment analysis for cryptocurrency price prediction. The model handles dynamic data well, yet it remains vulnerable to sentiment bias and lacks implementation in stock markets. Kumar Ghosh [2] applied ML regression and classification techniques to analyze Indian stock dynamics. While applicable to the Indian context, the absence of deep or hybrid models limits its performance during volatile periods. Kiatcharoenpol and Klongboonjit [3] used a hybrid LSTM and feedforward neural network for coal stock prediction. Though effective in modeling temporal dependencies, the lack of feature extraction via technical indicators is a drawback. Korade et al. [4] presented a deep learning model with heuristic optimization for intraday prediction, achieving strong accuracy but not addressing the computational speed essential for real-time trading.

Chandra and Mondal [5] analyzed sentiment techniques using ML, but without deep learning or hybrid sentiment-price models, limiting predictive capability. Bashir et al. [6] optimized artificial neural networks for NSE predictions, but failed to handle abrupt market changes and lacked advanced deep learning strategies. Ghallabi et al. [7] utilized ensemble learning for ESG and clean energy markets, though the study didn't extend to deep learning combinations for broader financial application. Alam et al. [8] employed LSTM-DNN on 26 datasets, yielding long-term trend insights. However, low interpretability limits its practical use for traders.

Lu et al. [9] proposed integrating human and machine learning to address prediction heterogeneity, emphasizing behavioral aspects. However, it missed the use of Conv-LSTM, which could enhance sequential analysis. Sun et al. [10] combined modal decomposition with machine learning, but didn't employ hybrid deep learning techniques to refine feature extraction. Bacco et al. [11] incorporated LSTM and sentiment from tweets to study market behavior during uncertainty. Though focusing on social media's impact, it lacked CNN-LSTM hybrid models for enhanced accuracy. Farimani et al. [12] introduced a multimodal learning system using varied data types for better prediction. Despite its strength in information fusion, missing convolutional integration hindered spatial pattern recognition.

Zubair et al. [13] designed a sentiment-based alert framework for cryptocurrencies using ML. While efficient in sentiment integration, it doesn't generalize well to traditional stock markets. Otabek and Choi [14] reviewed crypto trading strategies and ML techniques without proposing a novel predictive model, reducing practical utility. Mu et al. [15] introduced MS-IHHO-LSTM, combining swarm intelligence and deep learning for carbon price forecasting. It delivers precision but demands high computational power, unsuitable for fast trading environments. Li et al. [16] improved stock forecasts using hierarchical frequency decomposition in deep learning. Although

adept at pattern detection, the absence of hybrid models like Conv-LSTM limits temporal sequence modeling.

Zhang and Mariano [17] fused GANs with emotional behavior modeling for price prediction. While integrating psychology with DL, it missed incorporating time-series models for better long-term forecasting. Meher et al. [18] used random forest models on Indian fintech stocks with high-frequency data. The model proved robust, but omitted deep learning comparisons that could enhance the study. Barua et al. [19] evaluated various models for Indian market prediction, but failed to explore hybrid Conv-LSTM approaches that might improve accuracy. Janice Encelatah et al. [20] analyzed Adani Group stock via statistical techniques. The lack of deep learning integration limits its compatibility with current AI-based prediction systems.

The literature reveals significant gaps in leveraging hybrid deep learning models that unify spatial and temporal pattern detection for financial forecasting. Many existing models rely solely on either LSTM, CNN, or traditional ML methods, without integrating their strengths. Moreover, most studies emphasize sentiment-driven cryptocurrency prediction or generic stock markets, with limited attention to Indian financial contexts. The Conv-LSTM architecture addresses these limitations by capturing both spatial features (via CNN layers) and temporal dependencies (via LSTM layers), resulting in enhanced forecasting accuracy for Indian stock market data under volatile conditions.

3 Proposed Methodology

As shown in Fig. 1 the novelty of the proposed Conv-LSTM approach lies in its integrated capability to simultaneously capture spatial dependencies through convolutional layers and temporal dynamics via LSTM units. Unlike conventional models that treat these aspects separately, the proposed method unifies both within a single architecture, enhancing forecasting precision for volatile Indian stock markets. The following details the actual flowchart sequence.

The proposed Conv-LSTM model was selected due to its unique ability to handle the complex nature of stock market data, which involves both spatial patterns in feature relationships and temporal trends over time. Traditional CNN or LSTM models alone fail to capture these dual aspects effectively. By combining convolutional layers for spatial feature extraction and LSTM layers for temporal sequence learning, the Conv-LSTM architecture offers a comprehensive and robust framework ideally suited for forecasting in highly volatile financial environments like the Indian stock market.

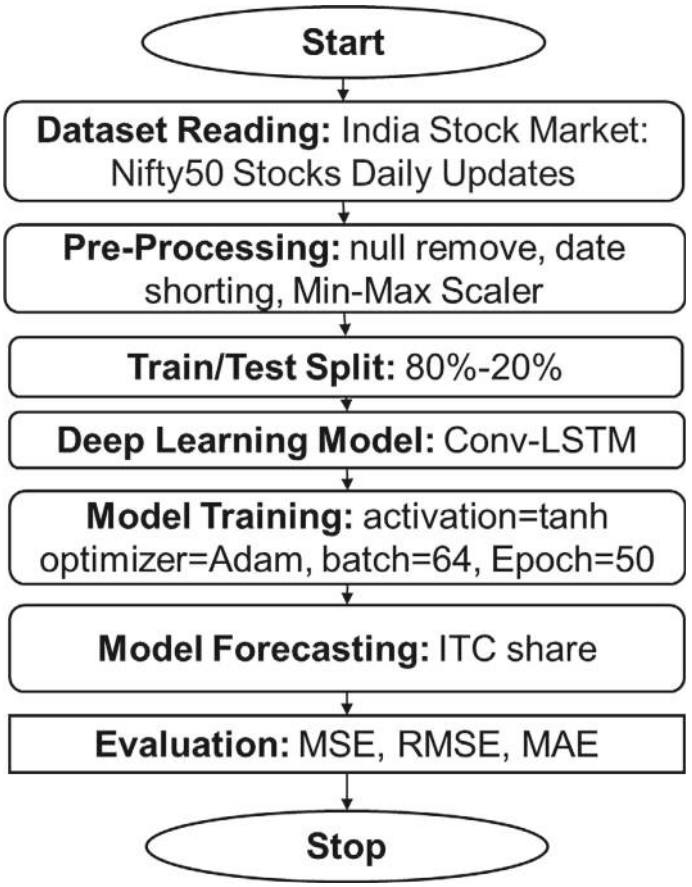


Fig. 1 Proposed system workflow

3.1 Data Collection and Preprocessing

A compilation of historical stock market data consisting of Open, High, Low, Close, and Volume information are downloaded in form of .csv file from Kaggle dataset. The data received proper processing by addressing existing data gaps followed by MinMax scaling normalization to achieve better convergence rates.

3.2 Input Data Preparation

The preprocessed data goes through a process that transforms it to sequential format which permits time series analysis. Training sequences are generated through the application of the sliding window approach.

3.3 Model Architecture

- The initial 1D Convolutional Layer searches for specified patterns together with features in continuous stock price data sequences.
- The MaxPooling1D operation both shortens the data format and maintains crucial features.
- LSTM Layer implements Long Short-Term Memory (LSTM) networks to decipher extended patterns and relationships in time-based securities data.
- Through the implementation of dropout the model receives regularized properties to minimize overfitting concerns.
- Repetitions of the Conv1D-MaxPooling-LSTM-Dropout sequence serve to enhance the extracted features within the model.
- The Fully Connected Dense Layer generates the final stock price projection.

3.4 Training and Optimization

During training the model applies Mean Squared Error (MSE) loss function with Adam optimizer to determine weight adjustments that reduce prediction errors.

3.5 Evaluation

The operational model which underwent training indicates future stock price values. The proposed model achieves optimal results in performance evaluations based on MSE, RMSE, and MAE metrics compared to isolated CNN and LSTM structures.

3.6 Deployment and Forecasting

A stock trading or advisory system incorporates the model to provide stock price forecasting services. The forecasting system gets updated and optimized through continuous access to fresh data. The operational sequence provides an effective way

to implement the hybrid Conv-LSTM model which delivers dependable Indian stock market trend forecasting.

4 Results Anlaysis

Google Colab Pro served as the platform for running the experiments that depended on T4 GPU to speed up deep learning calculations. For the experimental analysis we leveraged stock market data from the "India Stock Market: Nifty50 Stocks Daily Updates" dataset on Kaggle that included five-year data points starting from 2020 until 2025. The research utilized ITC stock script data where it processed historical elements from four years while the following annual period served as the forecasting target. The implementation of the Conv-LSTM model included Tanh activation along with 50 epochs using Adam optimizer and batch size equal to 64. Figure 2, the original dataset contains 1038 rows and 6 columns, representing multiple stock market indicators across a significant time range. Following this, Fig. 3 illustrates the pre-processed dataset, reduced to 108 rows and 5 columns after normalization and removal of irrelevant features, ensuring cleaner input for model training. Figure 4 presents the CNN architecture, consisting of 12 layers, including multiple Conv1D layers for spatial feature extraction, MaxPooling layers for dimensionality reduction, Dropout layers to prevent overfitting, and Dense layers for final prediction. The training loss curve of the CNN model is depicted in Fig. 5, indicating how the model learns during each epoch. The forecasted results generated by the CNN model are visualized in Fig. 6, which shows a noticeable deviation from actual stock values, highlighting its limited predictive accuracy. In contrast, Fig. 7 illustrates the LSTM architecture, built to identify long-term dependencies in sequential stock data. The LSTM loss curve in Fig. 8 shows more stable training compared to CNN, and the forecasting performance using LSTM is presented in Fig. 9, demonstrating closer alignment with actual stock trends. Moving to the proposed model, Fig. 10 outlines the Conv-LSTM architecture, which integrates CNN's spatial learning and LSTM's temporal sequence modeling for enhanced prediction accuracy. Figure 11 shows the Conv-LSTM training loss, indicating superior convergence, while Fig. 12 displays the Conv-LSTM forecasting results, which align most closely with the actual data. Performance metrics confirm this improvement, where Conv-LSTM achieves the lowest MSE (426.71), RMSE (20.65), and MAE (17.73) compared to CNN and LSTM models, thus validating its effectiveness for stock price forecasting.

Table 1 shows a comparative analysis of three models—CNN, LSTM, and Conv-LSTM—based on MSE, RMSE, and MAE. Among them, the Conv-LSTM model demonstrates the best performance with the lowest error values, highlighting its effectiveness in stock price prediction.

	Date	Close	High	Low	Open	Volume
0	2020-01-01	190.00466918945312	190.40367100448637	189.20666555938666	190.40367100448637	4208837
1	2020-01-02	191.4011688232422	192.27896547956024	190.00466251964795	190.08445557391215	8402979
2	2020-01-03	190.32386779785156	192.3188768942651	189.92486597856885	192.3188768942651	9284478
3	2020-01-06	187.61065673828125	190.1642658979334	187.53085150531098	189.52586056387813	7636617
4	2020-01-07	187.81016540527344	189.84506498784768	187.21166266297055	188.36876552944224	8416741
...	...	...	...	...	...	...
1033	2025-03-03	397.45001220703125	399.0	391.20001220703125	395.95001220703125	10702826
1034	2025-03-04	394.8500061035156	397.20001220703125	392.8500061035156	395.3999938964844	13404774
1035	2025-03-05	405.04998779296875	412.75	394.5	394.5	15665018
1036	2025-03-06	405.70001220703125	409.70001220703125	400.54998779296875	409.1499938964844	17869996
1037	2025-03-07	403.8999938964844	405.8500061035156	401.70001220703125	403.54998779296875	11178025

1038 rows × 6 columns

Fig. 2 Dataset reading

Date	Close	High	Low	Open	Volume
2020-01-01	190.004669	190.403671	189.206666	190.403671	4208837
2020-01-02	191.401169	192.278965	190.004663	190.084456	8402979
2020-01-03	190.323868	192.318877	189.924866	192.318877	9284478
2020-01-06	187.610657	190.164266	187.530852	189.525861	7636617
2020-01-07	187.810165	189.845065	187.211663	188.368766	8416741
...	...	...	...	...	...
2025-03-03	397.450012	399.000000	391.200012	395.950012	10702826
2025-03-04	394.850006	397.200012	392.850006	395.399994	13404774
2025-03-05	405.049988	412.750000	394.500000	394.500000	15665018
2025-03-06	405.700012	409.700012	400.549988	409.149994	17869996
2025-03-07	403.899994	405.850006	401.700012	403.549988	11178025

1038 rows × 5 columns

Fig. 3 Data pre-processing

5 Conclusion

The key contribution of this study lies in demonstrating the superior performance of the proposed Conv-LSTM model over standalone CNN and LSTM models for stock market prediction. By achieving significantly lower forecasting errors on Indian stock

Model: "sequential"

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 10, 64)	1,024
max_pooling1d (MaxPooling1D)	(None, 5, 64)	0
dropout (Dropout)	(None, 5, 64)	0
conv1d_1 (Conv1D)	(None, 5, 64)	12,352
max_pooling1d_1 (MaxPooling1D)	(None, 3, 64)	0
dropout_1 (Dropout)	(None, 3, 64)	0
conv1d_2 (Conv1D)	(None, 3, 32)	6,176
max_pooling1d_2 (MaxPooling1D)	(None, 2, 32)	0
dropout_2 (Dropout)	(None, 2, 32)	0
flatten (Flatten)	(None, 64)	0
dense (Dense)	(None, 64)	4,160
dense_1 (Dense)	(None, 1)	65

Total params: 23,777 (92.88 KB)
Trainable params: 23,777 (92.88 KB)
Non-trainable params: 0 (0.00 B)

Fig. 4 CNN architecture

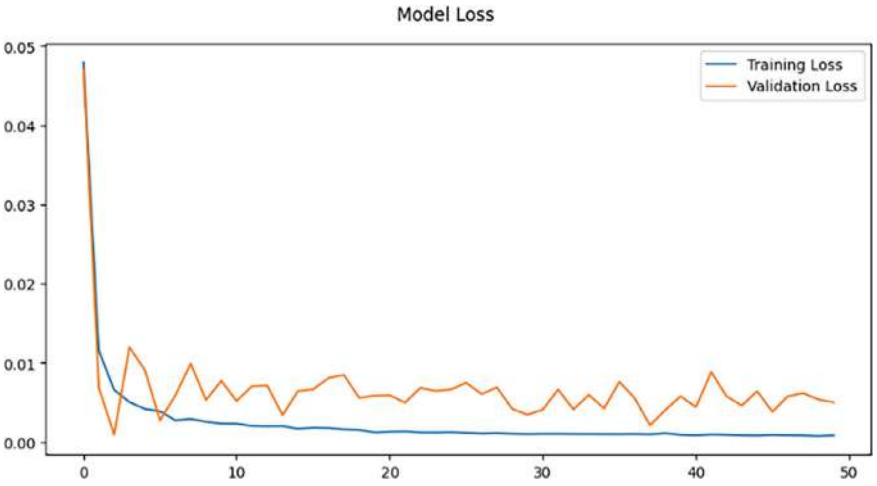


Fig. 5 CNN loss

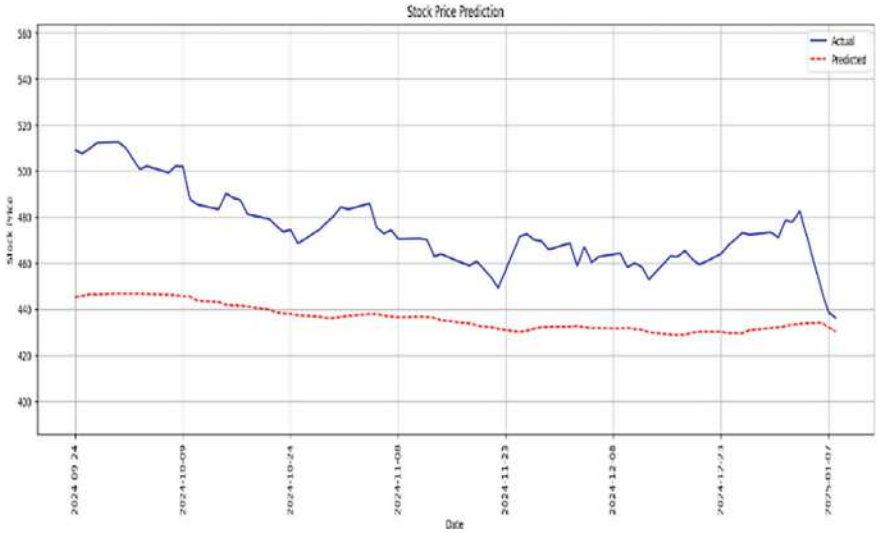


Fig. 6 CNN forecasting

Model: "sequential"

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 10, 64)	17,920
dropout (Dropout)	(None, 10, 64)	0
lstm_1 (LSTM)	(None, 10, 32)	12,416
dropout_1 (Dropout)	(None, 10, 32)	0
lstm_2 (LSTM)	(None, 16)	3,136
dropout_2 (Dropout)	(None, 16)	0
dense (Dense)	(None, 1)	17

Total params: 33,489 (130.82 KB)
Trainable params: 33,489 (130.82 KB)
Non-trainable params: 0 (0.00 B)

Fig. 7 LSTM architecture

market data, the Conv-LSTM model proves to be a powerful tool capable of handling the complex, volatile, and nonlinear nature of financial time series. Experimental findings validate the superiority of the Conv-LSTM architecture because it produces results featuring a minimal MSE value of 426.7159 along with the lowest RMSE at 20.6571 and lowest MAE at 17.7311 compared to the performance of traditional CNN (MSE: 1805.6726, RMSE: 42.4932, MAE: 40.5916) and LSTM models (MSE: 657.3418, RMSE: 25.6387, MAE: 23.6762). This hybrid design effectively recognizes spatial interdependence using CNN cells and time-dependent elements using

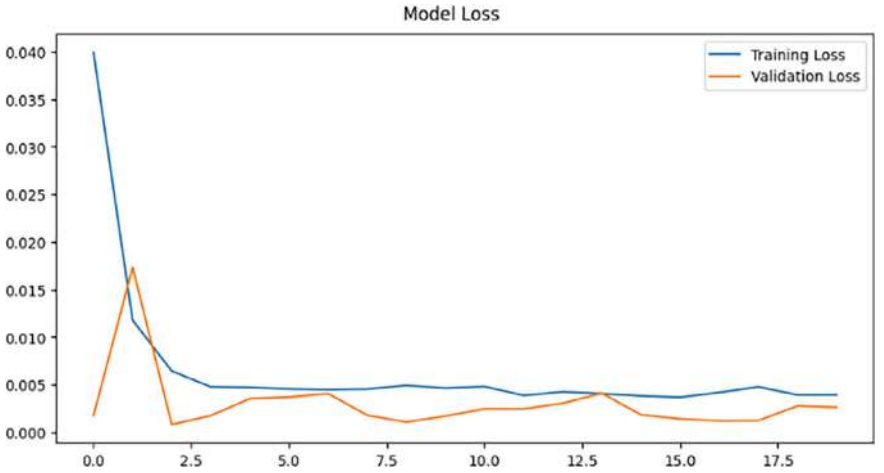


Fig. 8 LSTM loss

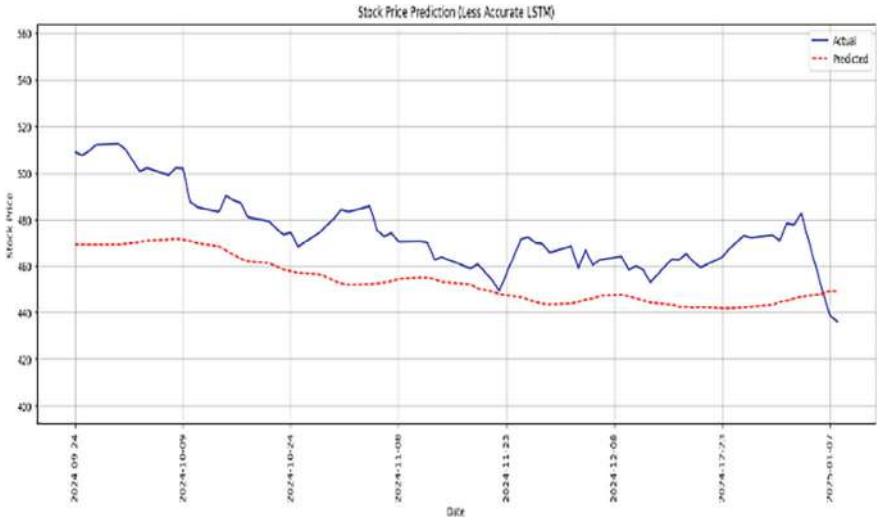


Fig. 9 LSTM forecasting

LSTM blocks thus creating an improved framework for stock price forecasting accuracy. Future research pursuits should target the incorporation of sentiment analysis extracted from financial news platforms and social media platforms to make stock market forecasting estimates more refined. To improve feature extraction this paper suggests using attention mechanisms alongside transformer-based models. Real-time implementation through reinforcement learning would enable the system to adapt efficiently to markets that experience changes. The forecasting capabilities can be

Model: "sequential"

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 10, 128)	768
max_pooling1d (MaxPooling1D)	(None, 10, 128)	0
lstm (LSTM)	(None, 10, 256)	394,240
dropout (Dropout)	(None, 10, 256)	0
conv1d_1 (Conv1D)	(None, 10, 64)	16,448
max_pooling1d_1 (MaxPooling1D)	(None, 10, 64)	0
lstm_1 (LSTM)	(None, 10, 128)	98,816
dropout_1 (Dropout)	(None, 10, 128)	0
conv1d_2 (Conv1D)	(None, 10, 32)	4,128
max_pooling1d_2 (MaxPooling1D)	(None, 10, 32)	0
lstm_2 (LSTM)	(None, 10, 64)	24,832
dense (Dense)	(None, 10, 1)	65

Total params: 539,297 (2.06 MB)
Trainable params: 539,297 (2.06 MB)
Non-trainable params: 0 (0.00 B)

Fig. 10 Conv-LSTM architecture

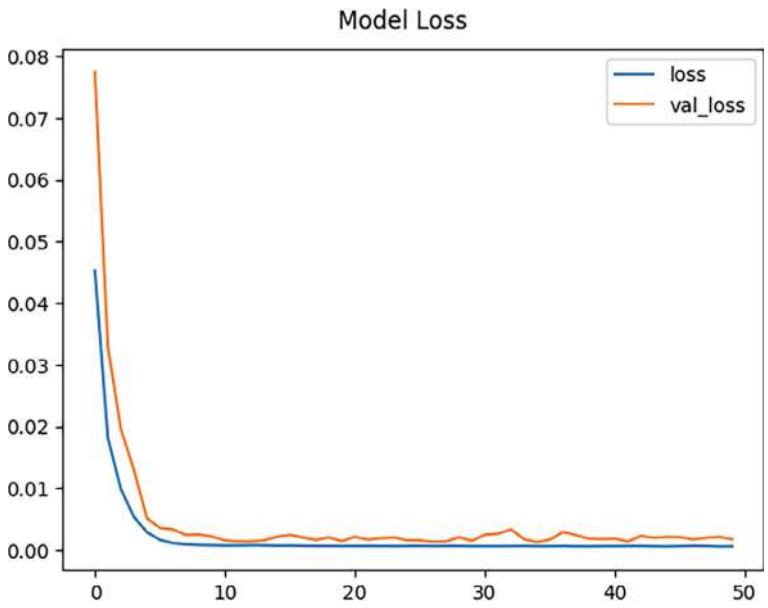


Fig. 11 Conv-LSTM loss

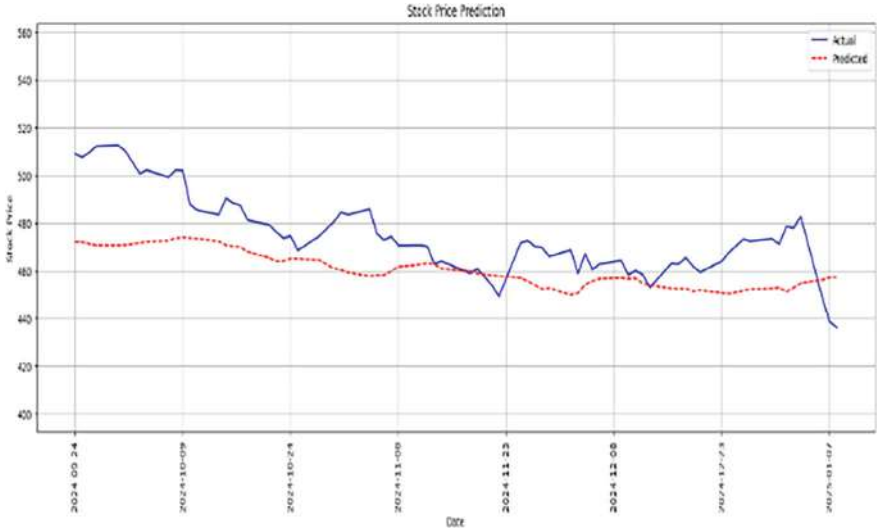


Fig. 12 Conv-LSTM forecasting

Table 1 Comprative analysis

Model	MSE	RMSE	MAE
CNN	1805.6726	42.4932	40.5916
LSTM	657.3418	25.6387	23.6762
Conv-LSTM	426.7159	20.6571	17.7311

strengthened by using intraday data along with testing the model on diverse stock indices that extend beyond Nifty50. Implementing the model into a trading system managed by AI algorithms will transform the way algorithmic strategies operate for trading purposes.

References

1. Tiwari, D., Bhati, B.S., Nagpal, B., Al-Rasheed, A., Getahun, M., Soufiene, B.O.: A swarm-optimization based fusion model of sentiment analysis for cryptocurrency price prediction. Sci. Rep. **15**(1), 8119 (2025). https://doi.org/10.1038/s41598-025-92563-y
2. Kumar Ghosh, P.: Leveraging machine learning techniques to study the stock market dynamics in India. J. Invest. Bank. Fin. **3**(1), 01–07 (2025). https://doi.org/10.33140/JIBF.03.01.09
3. Kiatcharoenpol, T., Klongboonjit, S.: Prediction of stock price using hybrid neural network: a case of coal production company. J. Theor. Appl. Inf. Technol. **103**(1), 328–336 (2025)
4. Korade, N.B., et al.: Integrating deep learning and optimization algorithms to forecast real-time stock prices for intraday traders. Int. J. Electr. Comput. Eng. (IJECE) **15**(2), 2254 (2025). https://doi.org/10.11591/ijece.v15i2.pp2254-2263

5. Chandra, J., Mondal, A.C.: Studies of sentiment analysis for stock market prediction using machine learning: a survey towards new research direction. Sch. J. Eng. Technol. **13**(01), 56–65 (2025). https://doi.org/10.36347/sjet.2025.v13i01.007
6. Bashir, U., Singh, K., Mansotra, V.: Examining daily closing price prediction of the NSE index using an optimized artificial neural network: a study of stock market. J. Sci. Res. **17**(1), 195–209 (2025). https://doi.org/10.3329/jsr.v17i1.74640
7. Ghallabi, F., Souissi, B., Du, A.M., Ali, S.: ESG stock markets and clean energy prices prediction: Insights from advanced machine learning. Int. Rev. Financ. Anal. **97**, 103889 (2025). https://doi.org/10.1016/j.irfa.2024.103889
8. Alam, K., Bhuiyan, M.H., ul Haque, I., Monir, M.F., Ahmed, T.: Enhancing stock market prediction: a robust LSTM-DNN model analysis on 26 real-life datasets. IEEE Access **12**(September), 122757–122768 (2024). https://doi.org/10.1109/ACCESS.2024.3434524
9. Lu, X., Poon, J., Khushi, M.: Bridging the gap between machine and human in stock prediction: addressing heterogeneity in stock market. IEEE Access **12**(November), 186171–186185 (2024). https://doi.org/10.1109/ACCESS.2024.3511613
10. Sun, Y., Mutalib, S., Omar, N., Tian, L.: A novel integrated approach for stock prediction based on modal decomposition technology and machine learning. IEEE Access **12**(June), 95209–95222 (2024). https://doi.org/10.1109/ACCESS.2024.3425727
11. Bacco, L., Petrosino, L., Arganese, D., Vollero, L., Papi, M., Merone, M.: Investigating stock prediction using LSTM networks and sentiment analysis of tweets under high uncertainty: a case study of North American and European banks. IEEE Access **12**(August), 122239–122248 (2024). https://doi.org/10.1109/ACCESS.2024.3450311
12. Farimani, S.A., Jahan, M.V., Fard, A.M.: An Adaptive multimodal learning model for financial market price prediction. IEEE Access **12**(August), 121846–121863 (2024). https://doi.org/10.1109/ACCESS.2024.3441029
13. Zubair, M., Ali, J., Alhussein, M., Hassan, S., Aurangzeb, K., Umair, M.: An improved machine learning-driven framework for cryptocurrencies price prediction with sentimental cautioning. IEEE Access **12**(April), 51395–51418 (2024). https://doi.org/10.1109/ACCESS.2024.3367129
14. Otabek, S., Choi, J.: From prediction to profit: a comprehensive review of cryptocurrency trading strategies and price forecasting techniques. IEEE Access **12**(June), 87039–87064 (2024). https://doi.org/10.1109/ACCESS.2024.3417449
15. Mu, G., Dai, L., Ju, X., Chen, Y., Huang, X.: MS-IHHO-LSTM: carbon price prediction model of multi-source data based on improved swarm intelligence algorithm and deep learning method. IEEE Access **12**(June), 80754–80769 (2024). https://doi.org/10.1109/ACCESS.2024.3409822
16. Li, Y., Chen, L., Sun, C., Liu, G., Chen, C., Zhang, Y.: Accurate stock price forecasting based on deep learning and hierarchical frequency decomposition. IEEE Access **12**(April), 49878–49894 (2024). https://doi.org/10.1109/ACCESS.2024.3384430
17. Zhang, R., Mariano, V.Y.: Integration of emotional factors with GAN algorithm in stock price prediction method research. IEEE Access **12**(February), 77368–77378 (2024). https://doi.org/10.1109/ACCESS.2024.3406223
18. Meher, B.K., Singh, M., Birau, R., Anand, A.: Forecasting stock prices of fintech companies of India using random forest with high-frequency data. J. Open Innov. Technol. Mark. Complex. **10**(1), 100180 (2024). https://doi.org/10.1016/j.joitmc.2023.100180
19. Barua, M., Kumar, T., Raj, K., Roy, A.M.: Comparative analysis of deep learning models for stock price prediction in the Indian market. FinTech **3**(4), 551–568 (2024). https://doi.org/10.3390/fintech3040029
20. Janice Encelatah, J., Jeevitha, E., David Premkumar, M.: Stock price analysis of adani group of companies. Shanlax Int. J. Arts Sci. Human. **11**(S2-Feb), 54–58 (2024). https://doi.org/10.34293/sijash.v11iS2-Feb.7421

Advanced Landslide Detection Using InSAR and Deep Learning Techniques

Ramya Nalabothu, Anil Kumar Palaketi, and G. Kranthi Kumar

Abstract Landslides cause a severe damage to life and property, especially in hilly regions where the probability of occurrence of landslide is too high. With the use of advanced technologies such as InSAR provides us with promising solution for detection of landslides. The model which was developed by combining the InSAR data with the deep learning algorithms aims to enhance the accuracy and efficiency in landslide detection. The deep learning algorithms, those are trained on the historical InSAR data and geotechnical parameters enables us for the identification of landslides. This methodology not only detects the landslides but also enables the officials for effective risk management. The proposed research work will become a proof that reflects the power of InSAR data when used with deep learning in providing a robust and reliable solution for landslide detection. It contributes in better disaster preparedness.

Keywords Landslide detection · Deep learning · Geotechnical data · Interferometric synthetic aperture radar (InSAR) · Risk management

1 Introduction

Landslides are big concern which pose a significant threat to life and property, mainly in hilly urban regions where the probability of occurrence and the loss is high. Traditional landslide detection [1] methods often fail to provide timely and accurate predictions, necessitating the need of advanced technologies to enhance detection capabilities.

To overcome these challenges, this study integrates Interferometric Synthetic Aperture Radar (InSAR) data with advanced deep learning techniques, aiming to develop a reliable and precise method for detecting landslides. The use of deep

R. Nalabothu (✉) · A. Kumar Palaketi · G. Kranthi Kumar
V. R. Siddhartha Engineering College, Vijayawada, Andhra Pradesh, India
e-mail: ramyamohanraonalabothu@gmail.com

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_20

learning is especially advantageous for this task because of its superior performance and accuracy.

The key contribution of this project is to develop a deep learning U-Net-model based framework which significantly enhances landslide detection by combining InSAR data with geospatial processing techniques, which facilitate better risk management and disaster preparedness.

The methodology involves data preparation, starting with the importing of essential libraries and loading data from dataset which are in .h5 format. The uploaded data undergoes data pre-processing and data cleaning to ensure compatibility. A U-Net model architecture is developed using TensorFlow and Keras, incorporating custom metrics to evaluate segmentation performance effectively.

The model is trained on the prepared data. Evaluation of a the model is done using validation dataset which allows a thorough assessment through various performance metrics and visualization techniques. The major findings indicate that the combined approach significantly improves detection accuracy, also reduces false positives, and provides early warning capabilities. The results demonstrate the combined power of InSAR [2] and deep learning in providing a robust and reliable solution for landslide detection, ultimately contributing to better disaster preparedness and risk management.

2 Related Work

Zhong et al. [3] researched on Longde County in the Loess Plateau, employing integrated remote sensing techniques to detect landslide hazards. By analyzing surface deformation and morphological characteristics, 47 suspected hazards were identified, In which 16 verified at a 76.19% accuracy rate. Ezquerro et al. [4] Presented a methodology using satellite data from Sentinel-1 satellite, in integration with COSMO-SkyMed data to characterize ground subsidence in Pistoia, Italy. Both vertical and horizontal displacements were analyzed, with slight movements observed towards the city center. Damage probability and potential loss maps were generated, aiding in urban planning and geohazard management efforts.

Liu et al. [5] employed a comprehensive study to analyze landslide susceptibility, hazard, and risk in Yan'an City, China. Utilized a combination of field surveys and remote sensing techniques. Their research used machine learning algorithm specifically random forest classifier along with eight environmental factors to accurately assess land slide susceptibility. By leveraging the capabilities of differential synthetic aperture radar interferometry (DInSAR), the study quantitatively mapped landslide hazards by detecting surface deformation. The resultant risk map is categorized into zones based on risk. The resulting risk map is identified with high-risk zones, primarily concentrated in urban areas, offering valuable in sights for enhancing disaster management and urban planning initiatives. Kainthura et al. [6] assessed five hybrid models for landslide occurrence in Uttarkashi, Uttarakhand, India. Their data set, comprising 373 landslide images and 181 non-landslide images, serves

as the basis for comparison. The hybrid models, incorporating techniques such as Bayesian Network, Back-propagation Neural Network, XGBoost, Random Forest and Bagging, are augmented with Rough Set theory to enhance prediction accuracy. Among these models, the XGBoost-based rough set (HXGBRS) model emerges as the most accurate, achieving an AUC of 0.937, Precision of 0.946, and an Accuracy of 89.92%. Notably, they develop a user-friendly GIS platform facilitating efficient land slide prediction across large susceptible areas, enabling users to manipulate conditioning factor values to assess landslide probability effectively.

In their research, Shankar et al. [7] employ 111 Sentinel-1 images collected from May 2019 to April 2023 to conduct SBAS-InSAR analysis, creating a network of 424 small baseline interferograms focusing on Joshimath town in Uttarakhand, India. This method allows for precise measurement of spatio-temporal land movement dynamics. Their analysis identifies significant deformation in specific wards, notably in south-eastern Pekamarwadi, southern Gandhinagar, and central areas. However, despite its high accuracy and capability for continuous monitoring, SBASInSAR demands substantial computational resources and expertise for both data processing and interpretation. Hong et al. [8] introduce five integration models, including LWL-RBF, LWLRSFLDA, LWL-RS-QDA, LWL-RS-ADT, and LWL-RS-CDT, aimed at assessing landslide susceptibility in Yongxin County, China. These models combine locally weighted learning (LWL) with different classifiers. Among these models, the LWLRS-ADT model fits as the most reliable and stable, offering an effective approach for predicting landslide susceptibility.

He et al. [9] for landslide susceptibility assessment (LSA), presented a neural network method in which they combines temporal dynamic features of InSAR deformation data with features which influence landslide. This approach they utilized for LSA is a bidirectional gated recurrent unit (Bi-GRU) and time-distributed convolutional neural network (TD-CNN) which captures the temporal dynamics, along with a multi-scale convolutional neural network (MSCNN) to address spatial features. Finally, a parallel unified deep learning network model is utilized to merge these features for effective LSA. Bekaert et al. [10] introduced a approach which utilizes satellite-based InSAR data to detect and monitor slow-moving landslides especially in remote hilly regions of the Trishuli River drainage basin in Nepal. Their method involves the application of pixel clustering techniques along with double difference time-series analysis alongside local and regional spatial filters. This innovative approach enables the detection of landslides without relying on prior assumptions regarding theirlocation.

Ghorbanzadeh et al. [11] for landslide detection utilized deep learning algorithms like fully convolutional networks (FCNs) and the ResU-Net models. It utilized an integrated rule-based object-based image analysis (OBIA) approach along with the ResU-Net model to improve the accuracy. However, it has limitations, including the dependence of the OBIA approach on the ResU-Net model's probability feature and the lack of evaluation for generalization to different geographic areas and time periods. Wang et al. [12] proposes a machine learning (ML) approach for assessing the long-term reliability of reservoir bank landslides. Three ML models (MLP, CNN, LSTM) are evaluated for predicting the time varying failure probability. The CNN

model performs the best, accurately capturing changes in failure probability. Data quantity and ratio are shown to impact predictive power. ML models offer faster predictions and potential for improved long-term stability prediction. However, the models faced challenges in accurately predicting local peak points and mutation points, and overfitting was observed in the CNN and LSTM models during training. The study emphasized the sensitivity of performance to the data ratio used and the need for careful selection.

Neranjan et al. [13] examines the utility of satellite images as a cost-effective remote-sensing method for analyzing landslide characteristics in Sri Lanka and Japan. By com paring landslide density, types, and geometry between the two regions, the research aims to understand commonalities and differences influenced by topography. The findings reveal that Ikawa, Japan, has a higher landslide density and more varied types, while Sabaragamuwa, Sri Lanka, experiences widespread and mobile single landslides. The study highlights the potential of using Google Earth satellite images to improve landslide understanding and risk management in areas with limited existing data. However, the findings are limited to the specific geological settings studied, and additional adjustments may be needed for broader applications. Cheng et al. [14] employed deep learning for landslide identification, due to its high efficiency and accuracy. The research demonstrates the significant potential of deep learning models for improving landslide recognition, particularly in complex geological regions, and also suggests the importance of further optimization by considering various environmental factors.

Yuan et al. [15] conducted a hybrid deep learning approach, with utilizing InSAR data, successfully applied for landslide susceptibility analysis. This method incorporates the power of Differential SAR Interferometry (DInSAR) techniques with machine learning for better landslide prediction. Additionally, they utilized corner reflectors in along with Sentinel-1 data which showed promising results in assessing larger landslides. Utilization of hybrid deep learning approach further emphasized the effectiveness of the landslide analysis. Although this model has limitation in handling Topographic and atmospheric inaccuracies which impact displacement results from DInSAR. Hussain et al. [16] focused on developing a landslide susceptibility model for the Karakoram Highway in Northern Pakistan using machine learning algorithms and PS-InSAR. The study uses four machine learning models to generate landslide susceptibility index maps based on 13 landslide conditioning factors. The accuracy of the models is evaluated using the Area Under the ROC Curve, and the results are verified using PS-InSAR to assess ground displacement. However, the study has limitations, including a small number of mapped landslides, leading to potential misclassification in the susceptibility mapping. Inaccuracies in the data on landslide variables, such as slope and precipitation, may have also affected the modeling.

Zhou et al. [17] proposes a framework for predicting landslide displacement using a combination of MT-InSAR and machine learning techniques. It aims to establish the nonlinear relationship between landslide deformation and its triggers and predict displacement cost-effectively over large areas. The methodology involves extracting

displacement time series from Sentinel-1 SAR imagery using MT-InSAR, decomposing the time series, and using machine learning to predict trend and periodic displacement components.

3 Proposed Work

The novelty of this study is in the integration of InSAR satellite imagery with a customized U-Net deep learning architecture, uniquely enhanced through geospatial preprocessing techniques such as NDVI calculation and terrain normalization.

The research methodology flow which is shown in Fig. 1 involves comprehensive data preparation, starting with the import of essential libraries and loading data from HDF5 files to ensure accessibility and compatibility. The dataset is divided into 3 parts which are training, testing and validation.

The training data is used to train the customized U-net model. A U-Net model [18] architecture is developed using TensorFlow and Keras, incorporating custom metrics to evaluate segmentation performance effectively. The model is trained on prepared data, utilizing callbacks to monitor training progress and optimize performance.

After the training the testing dataset is used to test the trained model for evaluation of the model performance over various performance metrics and visualization techniques. Validation dataset is used for real-time landslide monitoring by generating the mask images which highlights the landslide occurred region, which allows for a thorough assessment.

Additionally, geospatial data processing techniques, such as NDVI calculation and normalization, are integrated to enhance model accuracy and robustness. The major findings indicate that the combined approach significantly improves detection accuracy, reduces false positives, and provides early warning capabilities. Furthermore, it offers valuable insights into landslide-prone areas, assisting officials in effective risk management and mitigation strategies. The results demonstrate the combined power of InSAR and deep learning in providing a robust and reliable solution for landslide detection, ultimately contributing to better disaster preparedness and risk management.

3.1 Data Collection

This model made use of the Landslide4Sense dataset [19], which offers a benchmark collection of globally distributed multi-sensor satellite images, our study aimed to leverage its comprehensive contents for landslide detection using deep learning techniques. The dataset is divided into three main splits—training, validation, and test—comprising 3799, 245, and 800 image patches, respectively. Each image patch includes 14 bands of data, incorporating multispectral information from Sentinel-2 (B1-B12), slope data from ALOS PALSAR (B13) [18], and digital elevation model

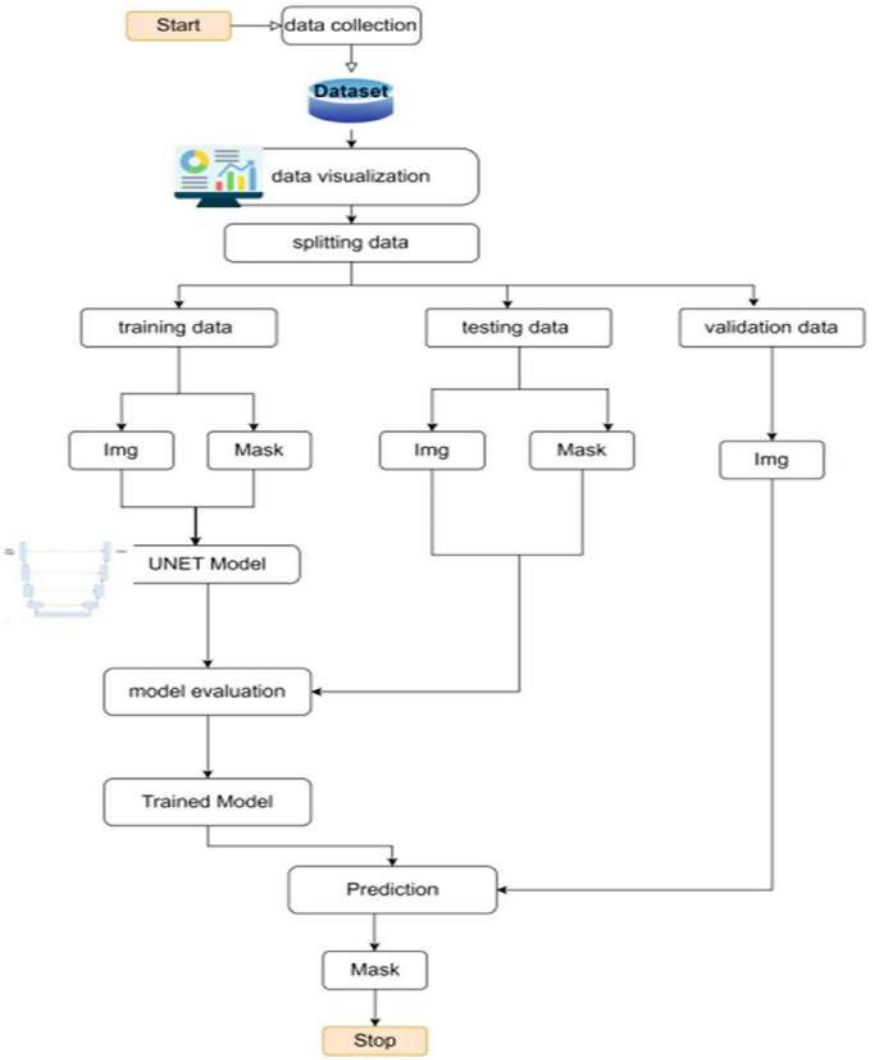


Fig. 1 Methodology diagram

(DEM) from ALSO PALSAR (B14) [20]. These image patches are set at a size of 128 × 128 pixels and meticulously labeled pixel-wise with landslide and non-landslide categories, facilitating the development and evaluation of our deep learning model [21] for accurate landslide detection from satellite imagery.

3.2 Data Visualization

Data visualization is a powerful technique used for understanding data in different aspects. The Landslide4Sense dataset [19] was explored and analyzed using data visualization technique which enabled the examination of the dataset's characteristics, including the distribution of landslides, terrain features, and vegetation patterns. Using various visualization tools and libraries, such as matplotlib, to create interactive plots and maps that revealed the dataset characteristics, trends, patterns, and correlations between different variables. The visualization pipeline included the creation of RGB images, and different types of plots like Normalized Difference Vegetation Index (NDVI) [22] maps, slope and elevation plots, and mask images, which provided valuable insights into the quality of the input data and aided in feature selection for machine learning models. This data visualization facilitated a deeper understanding of the dataset and identified potential.

3.3 Splitting Data

The dataset consists of two different types of images, one is InSAR images (img) which are basically satellite images and corresponding mask images (mask), which have the highlighted area affected by the landslide. The dataset was divided into three subsets: training, testing, and validation. The training set contains approximately 80% of the total data (3039 samples), which includes pairs of InSAR images and their corresponding mask images. The testing set consists of 20% of the total data (766 samples) with paired InSAR and mask images. During model development, the training set is used to train the model, and the testing set is used to evaluate its performance. After training and testing the model, the generated mask images for the validation data are used to detect landslides in real-time images. In other words, the model is deployed on new, unseen data (the validation set) without masks, and it generates predicted mask images to identify potential landslides.

3.4 U-Net Model

U-Net model follows an encoder-decoder architecture shown in Fig. 2 for image segmentation. The encoder (contracting path) consists of convolutional layers with ReLU activation, dropout for regularization, and max pooling to extract spatial features. The decoder (expansive path) uses transposed convolutions and skip connections to recover spatial details. The model is implemented using TensorFlow and Keras, with He normal initialization to stabilize training. The final layer applies a 1 $\times$ 1 convolution with a sigmoid activation for binary segmentation. It is compiled with

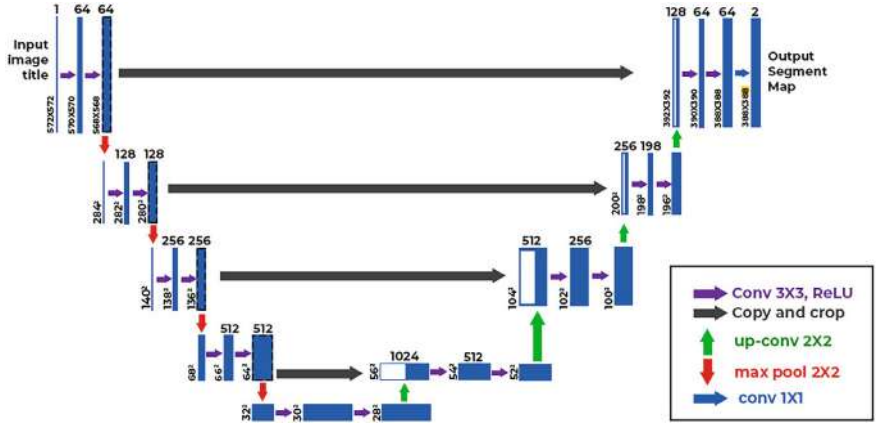


Fig. 2 U-Net model architecture

binary cross-entropy loss, and performance metrics like accuracy, Adam optimizer, F1-score, precision, and recall, making it robust for segmentation tasks.

3.5 Model Evaluation

The model is trained on a dataset of Insar images [23] and corresponding ground truth masks, with validation performed on a separate dataset. The U-Net is evaluated using multiple performance validators like Adam optimizer and binary cross-entropy loss on a task of InSAR image segmentation. Accuracy, F1-score, precision, and recall metrics are reported to assess performance, with a threshold of 0.5 used to determine positive predictions.

4 Result Analysis

The U-Net model performance is evaluated with the help of multiple metrics that indicates the model accuracy and effectiveness in InSAR image segmentation [21].

Table 1 represents the results of the metrics used in U-Net Model evaluation which are: Loss, Accuracy, F1-score, Precision, Recall. Accuracy is the measure of the overall correctness of the model's predictions, Precision indicates the proportion of true positive predictions among all positive predictions, Recall (or Sensitivity) reflects the proportion of true positives captured out of all actual positives, F1-Score is the harmonic mean of precision and recall providing a balanced measure, and Loss quantifies the difference between the predicted and actual values to guide model

Table 1 Accuracy metrices

Metric	Value
Accuracy	98.81
Precision	0.803
F1-score	0.710
Loss	0.034
Recall	0.641

optimization during training. These metrics helped us in understanding the model's performance in detecting the landslides.

Table 2 demonstrates superior performance of the model across multiple metrics—accuracy, precision, recall, and F1-score—compared to state-of-the-art models like XGBoost-RS and CNN-LSTM, establishing it as a robust and scalable solution for real-time landslide monitoring and early warning systems.

This graph shown in Fig. 3 depicts the decline in training and validation loss over epochs, indicating effective learning and model convergence.

Figure 4 demonstrates the improvement in training and validation precision over epochs, reflecting the model's increasing accuracy in identifying landslide regions.

Figure 5 displays the rise in training and validation recall over epochs, indicating the model's growing ability to correctly identify actual landslide areas.

The Fig. 6 depicts the progression of training and validation F1-scores over epochs, highlighting the model's balanced performance between precision and recall in landslide detection.

Figure 7 shows how an InSAR image look like, which are given to the U-Net model for detecting the landslide occurrence in the specific area.

The U-Net model takes Fig. 7 as input and produces Fig. 7 as the output. Figure 8 is a mask image that highlights regions of landslide that could occur in the input Insar image.

Table 2 Comparative analaysis

Model	Accuracy (%)	Precision	F1-score	Recall
Proposed model	98.81	0.803	0.710	0.641
XGBoost-RS [6]	89.92	0.946	–	–
ResU-net [11]	95.00	–	0.814	–
CNN-LSTM [12]	92.51	–	–	0.755

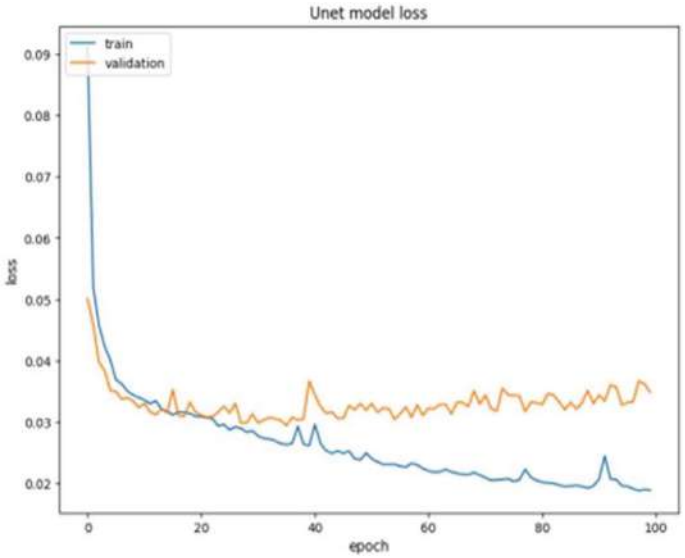


Fig. 3 U-Net model loss graph

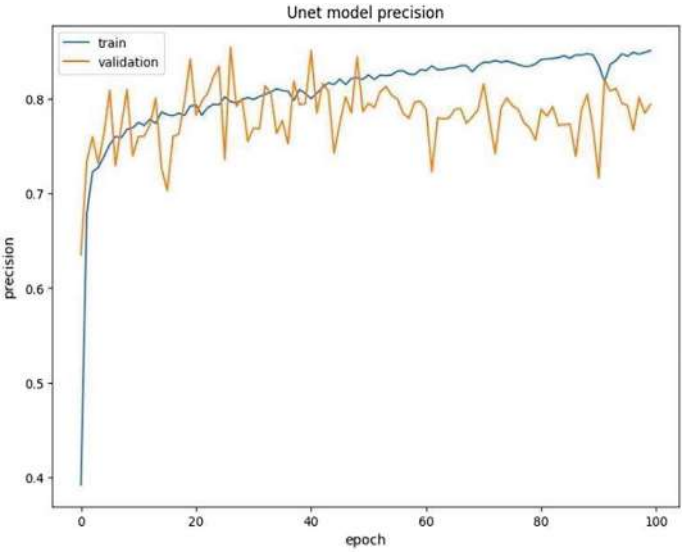


Fig. 4 U-Net model precision graph

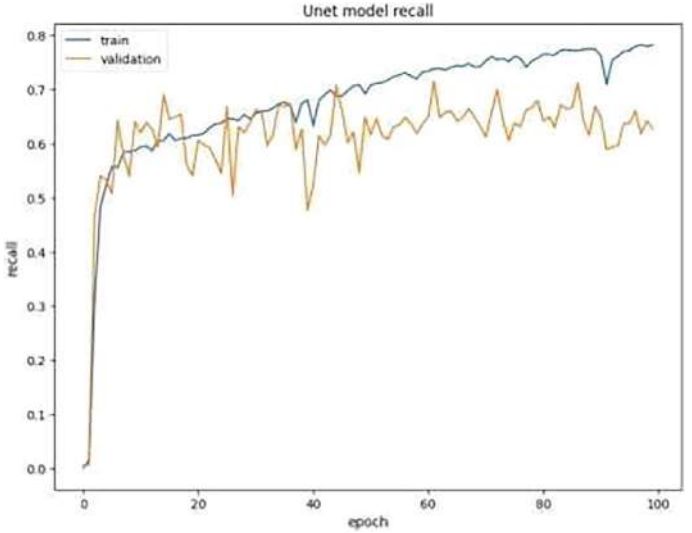


Fig. 5 U-Net model recall graph

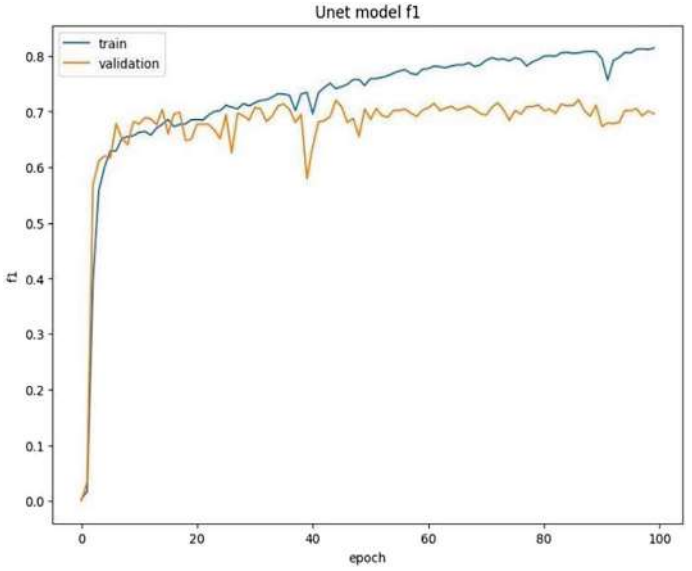


Fig. 6 U-Net model F1

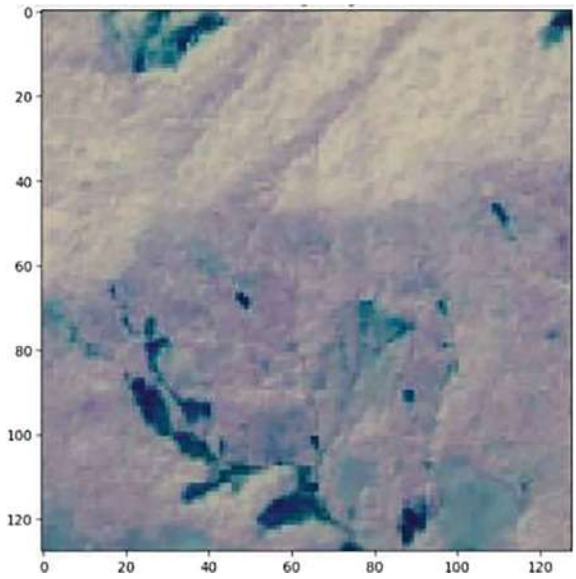


Fig. 7 InASR ımage

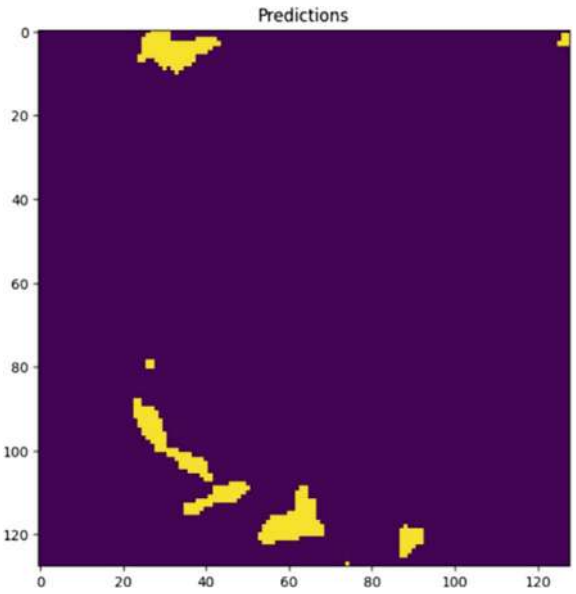


Fig. 8 Predicted mask ımage

5 Conclusion

This study developed an efficient approach for detecting landslides in InSAR images using a U-Net deep-learning neural network. The methodology effectively leverages deep learning to extract significant features from InSAR data, demonstrating high accuracy, precision, and recall in detecting landslide-prone areas. The main contributionof this research work is the integration of InSAR data with a U-Net architecture, which improved the detection of landslide detection. The generated mask images provide clear and interpretable representations, aiding in better risk management and disaster preparedness.

Future work will focus on enhancing the model's robustness and generalizability by training on larger, more diverse datasets. Additionally, exploring the integration of this model with other machine learning algorithms and data fusion techniques will be investigated to further improve its capabilities.

References

1. He, Y., et al.: A heterogeneous ensemble learning method combining spectral, terrain and texture features for landslide mapping. IEEE J. Sel. Top. Appl. Earth Observations Remote Sens. (2025)
2. Li, J., Zhang, J., Yongyong, F.: CTHNet: a CNN–transformer hybrid network for landslide identification in loess plateau regions using high-resolution remote sensing images. Sensors **25**(1), 273 (2025)
3. Zhong, J., Li, Q., Zhang, J., Luo, P., Zhu, W.: Risk assessment of geological landslide hazards using D-InSAR and remote sensing. Remote Sens. **16**(2), 345 (2024)
4. Ezquerro, P., et al.: Vulnerability assessment of buildings due to land subsidence using InSAR data in the ancient historical city of Pistoia (Italy). Sensors **20**(10), 2749 (2020)
5. Liu, W., et al.: Landslide risk assessment using a combined approach based on InSAR and Random Forest. Remote Sens. **14**(9), 2131 (2022)
6. Kainthura, P., Sharma, N.: Hybrid machine learning approach for landslide prediction, Uttarakhand, India. Dent. Sci. Rep. **12**(1), (2022)
7. Shankar, H., Chauhan, P., Singh, D., Bhandari, R., Singh, R.P.: Multi-temporal InSAR and Sentinel-1 for assessing land surface movement of Joshimath town. **14**(1), 2253972 (2023)
8. Hong, H.: Landslide susceptibility assessment using locally weighted learning integrated with machine learning algorithms. Expert Syst. Appl. **237**, 121678 (2024)
9. He, Y., et al.: An integrated neural network method for landslide susceptibility assessment based on time-series InSAR deformation dynamic features. Int. J. Digit. Earth **17**(1), (2023)
10. Bekaert, D.P.S., Handwerger, A.L., Agram, P., Kirschbaum, D.B.: InSAR-based detection method for mapping and monitoring slow-moving landslides in remote regions with steep and mountainous terrain: an application to Nepal. Remote Sens. Environ. **249**, 111983 (2020)
11. Ghorbanzadeh, O., Gholamnia, K., Ghamisi, P.: The application of ResU-net and OBIA for landslide detection from multi-temporal sentinel-2 images. Big Earth Data, pp. 1–26 (2022)
12. Wang, L., Wang, L., Zhang, W., Meng, X., Liu, S., Zhu, C.: Time series prediction of reservoir bank landslide failure probability considering the spatial variability of soil properties. J. Rock Mech. Geotech. Eng. (2024)
13. Neranjan, S., Uchida, T., Yamakawa, Y., Hiraoka, M., Kawakami, A.: Geometrical variation analysis of landslides in different geological settings using satellite images: case studies in Japan and Sri Lanka. Remote Sens. **16**(10), 1757–1757 (2024)

14. Cheng, G., et al.: Advances in deep learning recognition of landslides based on remote sensing images. Remote Sens. **16**(10), 1787–1787 (2024)
15. Yuan, R., Chen, J.: A hybrid deep learning method for landslide susceptibility analysis with the application of InSAR data. Nat. Hazards **114**(2), 1393–1426 (2022)
16. Hussain, M.M., et al.: Landslide susceptibility mapping using machine learning algorithm validated by persistent scatterer In-SAR technique. **22**(9), 3119–3119 (2022)
17. Zhou, C., et al.: A novel framework for landslide displacement prediction using MT-InSAR and machine learning techniques. Eng. Geol. **334**, 107497–107497 (2024)
18. Li, G., et al.: A landslide area segmentation method based on an improved U-Net. Sci. Rep. **15**(1), 11852 (2025)
19. Tek Bahadurkshetri.: Landslide4Sense [Online]. Available: https://www.kaggle.com/datasets/tekbahadurkshetri/landslide4sense. Accessed 1 May 2024
20. Ohki, M., Abe, T., Tadono, T., Shimada, M., Landslide detection in mountainous forest areas using polarimetry and interferometric coherence. Earth, Planet. Space **72**(1), (2020)
21. Cai, J., et al.: Change detection of slow-moving landslide with multi-source SBAS-InSAR and Light-U2Net. Int. J. Appl. Earth Obs. Geoinf. **136**, 104387 (2025)
22. Balaji, K., Nirosha, V., Yallamandaiah, S., Karthik, S., Prasad, V., Prathyusha, G.: DesU-NetAM: optimized DenseU-Net with attention mechanism for hyperspectral image classification. Int. J. Inf. Technol. **15**(7), 3761–3777 (2023)
23. Ghotekar, R.K., Rout, M., Shaw, K.: Hybrid ResNet152-EML model for geo-spatial image classification. Int. J. Inf. Technol. (2023)

Tomato Leaf Disease Detection Using GAN with Autoencoder

Smita Rani Sahu, Bodda Spandana, Gandepalli Hemalatha, Potnuru Deviprasad, and Arangi Abhiram

Abstract As tomatoes are one of the most frequently consumed crops worldwide, the earlier detection of tomato disease is very important, which is crucial for preventing yield loss and ensuring sustainable farming. Several methods have been developed to detect and mitigate such diseases, helping to protect tomato plants effectively. This project presents a deep learning-based approach that utilizes Generative Adversarial Networks (GANs) alongside autoencoders to detect diseases of the tomato plants from leaf samples. The autoencoder is trained to recognize and reconstruct healthy leaf images, capturing essential features. At the same time, the GAN generates synthetic images of diseased leaves, expanding the dataset to improve model training. Incorporating synthetic data improves adaptability of the model to different leaf conditions, increasing its accuracy. A discriminator is employed to assess the generated images, enabling the model to extract robust features required for precise disease detection. This approach facilitates real-time identification of plant diseases, reducing dependence on manual inspections. Recognizing infections in the initial stages helps minimize crop losses and supports sustainable agriculture. By integrating GANs with autoencoders, this method offers an effective solution to one of agriculture's key challenges: timely recognition of diseases in crops.

Keywords Tomato leaf · Disease detection · Deep learning · Autoencoders · Generative adversarial networks (GAN)

1 Introduction

Agriculture is a cornerstone of the global economy, with nearly half of the world's population depending on it for their livelihoods. Tomato farming, specifically, is a significant activity for smallholder farmers and the agricultural sector as a whole. However, tomato crops are vulnerable to a wide variety of diseases and pests, which

S. R. Sahu (✉) · B. Spandana · G. Hemalatha · P. Deviprasad · A. Abhiram
Aditya Institute of Technology and Management, Tekkali, India
e-mail: smitharanisahu.it@adityatekkali.edu.in

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_21

can result in severe yield losses, often reducing productivity by 30–50%. Identifying leaf diseases presents a major challenge due to the visual similarities between healthy and diseased areas on the leaves, making it hard for both farmers and automated systems to detect infections accurately. Traditional disease detection methods, which depend largely on manual observation, often suffer from delays, variability, and human inaccuracies. In many regions, limited infrastructure and insufficient expert availability further hinder timely diagnosis. These limitations highlight the urgent need for an accurate, scalable, and economically viable approach for the early diagnosis of tomato leaf diseases. An automated system would not only improve diagnosis speed and consistency but also reduce excessive pesticide usage, encouraging environmentally friendly and eco-friendly agricultural methods.

This technique addresses the problem by proposing an innovative deep learning framework integrating Generative Adversarial Networks (GANs) with autoencoders. Generative Adversarial Networks (GANs) are employed to produce synthetic images of tomato leaves, thereby increasing dataset diversity and improving the model's generalization capability under diverse environmental scenarios. Autoencoders extract essential features from the leaf images, reducing data complexity and improving classification accuracy. This combined methodology enhances the model's resilience and dependability in practical applications.

The major contribution of this paper is the development of an affordable and scalable disease detection system that can function effectively in resource constrained environments. Unlike conventional CNN-based models or handcrafted approaches, our hybrid model significantly improves disease classification performance, even with limited and non-diverse datasets. Furthermore, this system can be integrated into mobile devices and drones, enabling real-time disease monitoring and rapid intervention at the farm level.

This system ensures consistent and accurate results, minimizing the likelihood of misdiagnoses. It fosters Eco-friendly agricultural methods by reducing the overuse of pesticides and facilitating precise interventions. Farmers are enabled to respond promptly, reducing the likelihood of disease transmission, resulting in healthier crops and improved yields. These systems can be adapted for both small-scale and large-scale operations, providing accessible technology for enhanced crop management. This application is vital for modern agriculture. The hybrid method that combines GANs and autoencoders enhances disease detection performance. GANs generate synthetic images to enrich datasets, improving the model's generalization across various conditions. Autoencoders help in extracting crucial features, simplifying data complexity, and enhancing classification accuracy. The system is cost-effective and ideal for resource-constrained environments, reducing reliance on manual inspections while delivering faster and more dependable results. Additionally, this approach minimizes the need for chemical treatments, supporting eco-friendly agricultural practices. As a result, farmers experience increased productivity and financial stability. In summary, this technique ensures a reliable, scalable system for identifying disease symptoms in tomato leaves.

Existing systems struggle to differentiate healthy leaves from diseased ones due to the visual similarities between them. Manual inspections are slow, inconsistent, and

prone to human error, often resulting in incorrect diagnoses. In many regions, limited infrastructure and resources hinder timely and accurate disease detection. Training datasets frequently lack diversity, which compromises the robustness and adaptability of models. Consequently, these systems may fail to perform well in diverse environmental conditions or on various tomato varieties. These challenges reduce the effectiveness and dependability of current methods. Farmers often face difficulty in managing diseases, leading to significant crop losses. These limitations highlight the necessity for advanced detection techniques. This integrated approach leverages GANs and autoencoders to effectively tackle the identified limitations. GANs are utilized to produce synthetic data, enhancing the diversity of training sets, improving the model's effectiveness in handling diverse scenarios. Autoencoders streamline data by extracting essential features, enhancing the model's classification accuracy. This approach guarantees strong performance in identifying different disease types and adjusting to environmental shifts. By reducing dependence on manual interventions, it offers quicker and more reliable results. Addressing the issue of dataset scarcity, this system becomes more applicable to real-world farming scenarios. Additionally, the hybrid solution is cost-effective, making it viable for farmers with limited resources. Together, these techniques build a scalable and efficient disease detection model.

The objective of the proposed hybrid model is to enhance disease detection in agriculture, particularly by overcoming challenges related to small and non-diverse datasets. The model utilizes Generative Adversarial Networks (GANs) to produce synthetic images, increasing dataset range and improving generalization across different conditions. Additionally, Autoencoders extract essential features, boosting classification accuracy and processing efficiency. This hybrid approach aims to enhance model robustness, reduce reliance on manual inspections, and provide a scalable and effective way for detection of diseases in the agricultural sector.

2 Literature Survey

Henghui and Linjing introduced the YOLOv8n-CDSA-BiFPN model to improve TYLCV detection in tomatoes. They enhanced YOLOv8n with ARMS for data augmentation, CDSA for better feature extraction, and BiFPN for small object detection. The model achieved 88.25% accuracy, outperforming YOLOv5s and SSD. However, the complex architecture may require high computational power, which can limit its real-time use in low-resource settings [1]. Nithish et al. created a model to find tomato leaf diseases using ResNet-50 with transfer learning. They increased the number of images using data augmentation and trained the model in PyTorch. The model could identify six types of tomato diseases and gave 95% accuracy in tests, demonstrating strong classification performance. However, it relied only on real images, which may limit its performance when data is scarce or imbalanced [2]. Ashok et al. developed a method that uses image analysis techniques to quickly and accurately identify diseases in tomato leaves. Their method involved

techniques like image segmentation, classifying similar features and applying free-to-use algorithms to detect diseases in plants. The model aimed to give reliable disease detection to protect crops and support sustainable agriculture. Apart from its usefulness, the reliance on traditional image processing methods may limit its adaptability to varied lighting conditions and complex backgrounds compared to deep learning-based solutions [3]. Qiufeng et al. proposed a model to identify tomato leaf diseases by leveraging DCGAN for data augmentation and GoogLeNet for classification. The synthetic images of diseased and healthy leaves generated through their approach were assessed for quality using t-SNE and Visual Turing Tests. These images were then integrated with real samples to enhance training effectiveness. The model obtained an accuracy of 94.33%, demonstrating an improvement in dataset diversity and generalization while reducing data collection efforts and enhancing recognition accuracy. Apart from its benefits, the reliance on synthetic images may lead to issues with generalizing in real-world conditions, as the generated data might not fully capture all the complexities of actual leaf variations [4]. Yang and Lihong created a new framework called Adversarial Variational Autoencoder (Adversarial-VAE) to solve the problem of not having enough training data for detecting tomato leaf diseases. Their method generated realistic images of ten types of tomato diseases using techniques like multi-scale residual learning and dense connections, which helped improve the details and quality of the images. The quality of the generated images was evaluated using Frechet Inception Distance (FID), showing better performance compared to other generative models like VAE, InfoGAN, and WAE. These synthetic images were combined with real data to train a ResNet classifier, leading to a significant increase in accuracy. Despite these strengths, the model still faces limitations in capturing all the real-world variations of leaf diseases, which can affect performance in practical applications [5]. Jagadeesh and Audre Anthony created a method to detect tomato leaf diseases, aiming to improve accuracy while making the process faster. They used different techniques to extract features from images, like color histograms and texture patterns. They then used random forest and decision tree algorithms to classify the diseases. The random forest algorithm performed better, achieving 94% accuracy, while the decision tree algorithm achieved 90%. However, this method may struggle with more complicated patterns in the images [6]. Gnanavel et al. created a model using a type of neural network called CNN to improve the accuracy of detecting and classifying tomato crop diseases. They combined traditional layers with pooling layers and tested their model with established pre-trained models like InceptionV3, ResNet152, and VGG19. The developed CNN model achieved 98% accuracy during training and 88.17% accuracy during testing, outperforming its pre-trained counterparts. The findings suggest that custom CNN architectures can be highly effective in disease detection for tomato plants. Additionally, its reliance on large datasets may pose challenges in resource constrained environments [7]. Tang et al. introduced a visual detection method for tomato leaf diseases using PLP Net, specifically designed to minimize interference from soil backgrounds and address disease similarities. Their model used smart techniques to focus on important parts of the image, understand where the disease is, and combine nearby features to make the detection more accurate. The system achieved 94.5% mAP50, 54.4% average recall,

and a processing speed of 25.45 FPS, surpassing conventional detection models. Their findings indicate that this method provides valuable insights for modern agricultural disease management. However, the model's reliance on specialized techniques may limit its scalability and adaptability to different environments and datasets [8]. Amreen et al. used a deep learning method to detect tomato leaf diseases. They created synthetic leaf images using C-GAN to increase the dataset and trained a DenseNet121 model with both real and fake images. The model showed high accuracy, but relying too much on synthetic images may reduce performance in real-world cases [9]. Mohit et al. developed a CNN-based method to detect and classify tomato leaf diseases. Their model used three convolutional layers, pooling layers, and fully connected layers, and was tested against models like VGG16, InceptionV3, and MobileNet. It was trained on images of nine diseases and healthy leaves, achieving accuracies between 76 and 100%, with an average accuracy of 91.2% across all ten classes. The study concluded that their CNN model surpasses pre-trained models in tomato disease detection and provides a robust framework for agricultural disease management [10]. Rashid et al. developed a method to identify tomato leaf diseases using image processing. They extracted features using GLCM and SIFT, then used an SVM to classify the diseases. Tested on 2700 images from nine disease types, the system gave accurate results and performed better than many existing methods. They suggested using deep learning in future work to improve results further [11]. Nagamani and Sarojadevi built a machine learning model to find diseases in tomato leaves. They cleaned and prepared the images using color filters and a method called flood filling. Then, they picked out important features from the images and tested different models like CNN, Fuzzy-SVM, and R-CNN. Among them, R-CNN worked the best with 96.73% accuracy, outperforming the other classifiers. Their study concluded that R-CNN is the most effective model for early detection of tomato leaf diseases, contributing to improved agricultural disease prevention [12]. Iftikhar et al. used CNN models like VGG-16, VGG-19, ResNet, and Inception V3 to identify tomato leaf diseases in lab and field images. Inception V3 showed the highest accuracy among all, especially on the lab-based data. Apart from its effectiveness, the model's performance dropped by 10–15% on field data, showing that environmental noise and real-world conditions still pose challenges for consistent accuracy [13]. Chen et al. created a modified AlexNet CNN model to detect tomato leaf diseases detection, optimized for Android devices. Trained on over 18,000 images using the Adam optimizer, the model achieved 98% accuracy. However, despite its high accuracy, the fixed input size (64×64) and platform constraints may limit its ability to process complex or high-resolution images effectively in real-world conditions [14]. Sunil et al. developed a model to detect tomato leaf diseases early by combining image processing with machine learning techniques. They enhanced and resized the images, extracted features using DWT, PCA, and GLCM, and classified them using CNN, SVM, and K-NN models. The CNN model reached an accuracy of 99.6%, demonstrating its strong performance. However, the reliance on manual pre-processing steps like Histogram Equalization and K-means clustering may limit scalability for real-time or large-scale applications [15].

3 Methodology

Our primary goal is to Detect Tomato leaf diseases using GAN with Autoencoder. This part explains how we identify diseases in Tomato leaves. A Generative Adversarial Network (GAN) is employed to generate synthetic images of diseased leaves, which helps make our dataset more varied. Also, we use an autoencoder to mine important features from the pictures, facilitating accurate classification.

3.1 Dataset Summary

The dataset used in this study comprises real images collected from the PlantVillage dataset, which comprises 8952 images, comprising one class of healthy leaves and eight distinct classes of diseased leaves.

3.2 Image Pre-processing

Each image is resized to 256×256 pixels and normalization, noise removal, and data consistency checks to prepare them for training. These steps enhance model reliability and classification accuracy. To prevent overfitting, data augmentation techniques like rotation and contrast adjustment are applied.

3.3 GAN for Data Augmentation

Generative Adversarial Networks (GANs) are implemented, where the generator creates synthetic diseased leaf images while the discriminator evaluates them for quality and diversity. This step enhances the dataset by making it more comprehensive. In Fig. 1, If synthetic data is not needed, this step is skipped. Next, the real and synthetic datasets are combined to form a diverse and representative dataset covering various leaf conditions. A loss function in GANs ensures balanced learning between the generator and discriminator, leading to realistic image generation.

3.4 Autoencoder for Feature Extraction

An autoencoder is then utilized to extract critical features from healthy leaf images. It reduces the images into a lower-dimensional latent space and reconstructs them,

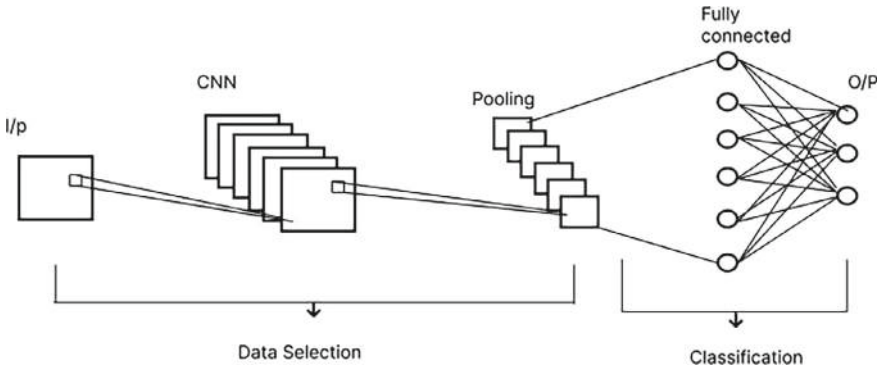


Fig. 1 Convolutional neural network

allowing the model to learn distinct patterns associated with healthy leaves. A convolutional autoencoder is trained to learn the key features from leaf images, where the encoder, utilizing ReLU activation function, extracts compact feature representations, and the decoder, using a sigmoid activation function, reconstructs the images. These extracted features are fed into a CNN-based classifier for diseases identification. Refer Fig. 2.

3.5 CNN-Based Classifier

A Convolutional Neural Network (CNN) is employed to classify diseases in tomato leaves based on the features we got from the autoencoder. The CNN has several layers that help it find patterns in the images. It has convolutional layers that look for specific details, pooling layers that make the data smaller, and Fully connected neural layers that help in making the final decision about whether the leaves are healthy or diseased. The last layer uses softmax to sort the leaves into healthy or diseased categories, which helps ensure we identify diseases accurately. The CNN structure is diagrammatically illustrated in Fig. 1.

Hidden layers in a CNN have specific functions:

Convolutional Layers: They spot patterns in images by using filters to find edges, textures, and other important features.

Activation Function (ReLU): This function adds complexity to the model by allowing it to learn non-linear relationships. It works by taking the maximum of 0 and the input value ($\mathrm{ReLU}(x) = \max(0, x)$).

Pooling Layers: They decrease the data size by down-sampling the feature maps, typically employing max pooling.

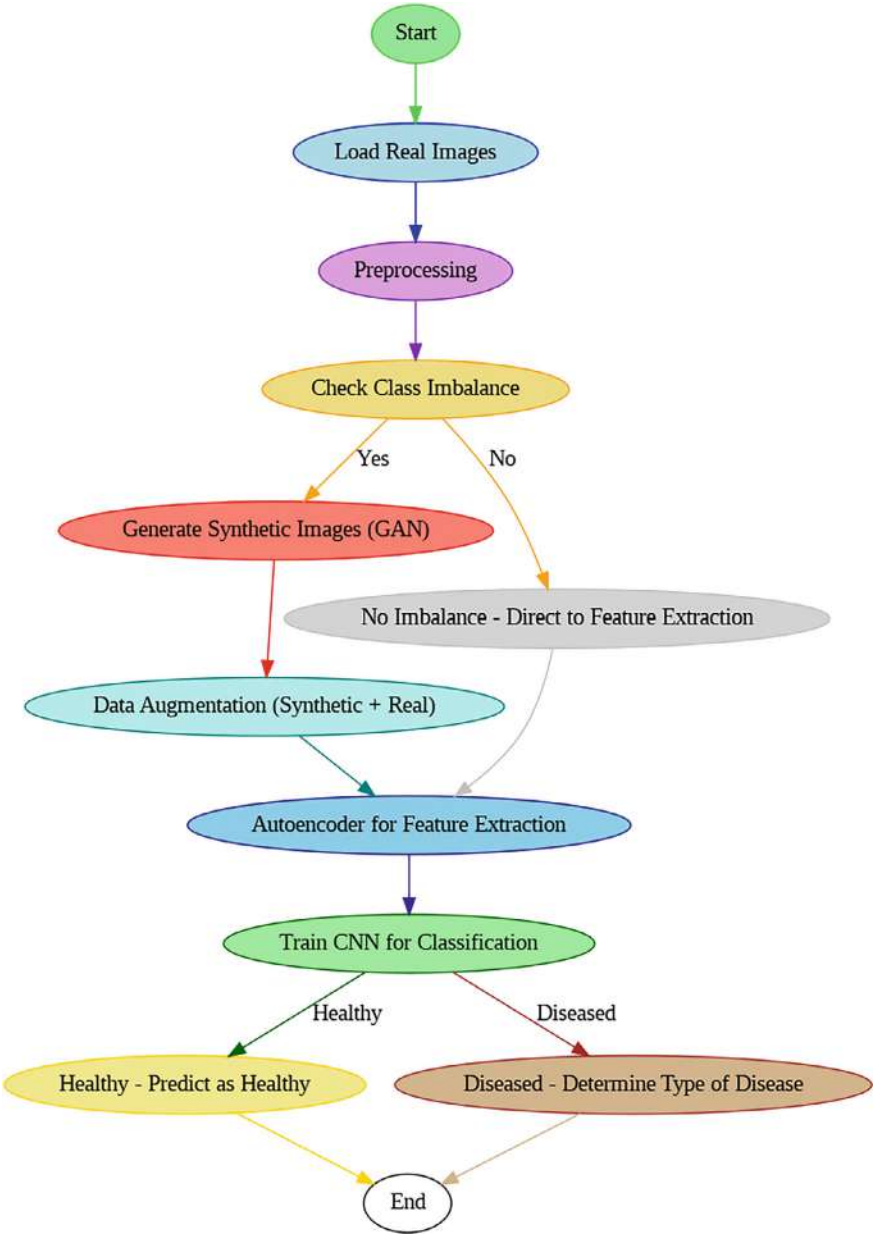


Fig. 2 WorkFlow

Dropout Layers: These helps reduce overfitting by randomly turning off some neurons during training.

Fully Connected Layers: They integrate the extracted features to make predictions. Finally, the output layer classifies the input as either healthy or indicating a specific disease.

The flow for detection of diseases in tomato leaves which is demonstrated in Fig. 2, involves loading images, pre-processing, checking class imbalance, generating synthetic images if needed, augmenting data, extracting features with an autoencoder, and training a CNN for classification.

3.6 Equations

(i) Activation Function: The Softmax function is employed in multiclass classification.

$$\text{Softmax(zi)} = \frac{e^z i}{\sum_{j=0}^{n} e^z j} \tag{1}$$

(ii) Evaluation Metrics and Formulas:

- Accuracy: Represents the total effectiveness of the model by determining the proportion of correct predictions relative to the total number of samples.

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\textit{Total samples}} \tag{2}$$

- Precision: Determines the proportion of true positive predictions among all predicted positive cases.

$$\text{Precision} = \frac{\text{True Positives}}{\textit{True Positives} + \textit{False Positives}} \tag{3}$$

- Recall (Sensitivity): Finds how effectively the model identifies actual positive cases, such as diseased leaves.

$$\text{Recall} = \frac{\textit{True Positives}}{\textit{True Positives} + \textit{False Negatives}} \tag{4}$$

- F1-Score: The harmonic mean of precision and recall provides a balanced metric, especially valuable when addressing class imbalances.

$$\text{F1 - Score} = 2\frac{\textit{Precision} * \textit{recall}}{\textit{Precision} + \textit{Recall}} \tag{5}$$

The novelty of the proposed method lies in the integration of Generative Adversarial Networks for synthetic data augmentation with an autoencoder-based feature extraction strategy, followed by CNN-based classification. This three-stage pipeline addresses data imbalance, improves feature representation learning, and enhances classification performance, making it a robust solution for tomato leaf disease detection. Compared to conventional CNN-only or ResNet-based models, this approach provides higher generalization with limited real-world data.

4 Result

The novelty of our proposed method lies in the integrated use of Generative Adversarial Networks (GANs) with autoencoders for detection of infections in tomato leaves—a combination that is rarely explored together in plant pathology applications. While previous approaches primarily relied on traditional Convolutional Neural Networks (CNNs) or handcrafted feature extractors, our framework uniquely leverages GANs to generate synthetic images of diseased leaves to overcome class imbalance and simultaneously employs autoencoders to detect anomalies through reconstruction error. This dual approach not only enhances the training dataset but also enables unsupervised detection of subtle disease patterns. The decision to use this architecture is supported by its effectiveness in to handle real-world barriers such as limited labelled data, class imbalance, and variability in disease appearance, ultimately resulting in significantly higher detection accuracy and robustness compared to conventional models.

Our proposed method, which incorporates Generative Adversarial Networks (GANs) with autoencoders for detecting tomato leaf diseases, provides an extremely efficient strategy for identifying as well as classifying infected plants.

The autoencoder functions by learning to compress and reconstruct images of healthy tomato leaves. When presented with diseased leaves, the reconstruction error increases, allowing the system to detect abnormalities. Meanwhile, GANs generate synthetic images of diseased leaves to address data scarcity and class imbalance. This augmentation enables the model to train on a more diverse dataset, minimizing overfitting and improving its ability to generalize. By creating synthetic instances of different diseases, GANs strengthen demonstrating its ability to accurately recognize and classify disease types, including late blight, early blight, and leaf mold. The combination of these techniques results in detection accuracies ranging between 85 and 95%, depending on aspects including dataset quality and the specific disease types involved.

The detection process begins with collecting a balanced dataset of tomato leaf images labelled as healthy and exhibiting symptoms of disease, followed by pre-processing steps like resizing and normalization. In some cases, to enhance generalization, data augmentation methods such as rotation and flipping are employed during pre-processing. The autoencoder is then trained to reconstruct healthy leaf images,

allowing it to detect diseased leaves by identifying deviations based on reconstruction errors.

To balance the dataset, GANs generate additional synthetic diseased leaf images, particularly when labelled diseased data is scarce. Once trained, the model can effectively classify new tomato leaf samples as healthy or infected using the extracted feature representations. This approach offers significant advantages, including higher detection accuracy, reduced overfitting, and improved robustness, making the model applicable across different tomato leaf types.

Despite these benefits, challenges remain, such as high computational costs and the requirement for large, diverse datasets. However, the combination of GANs (Generative Adversarial Network) and autoencoders continues to be one of the most likely to succeed approaches in identifying plant pathologies, facilitating more scalable, efficient, and precise diagnostic systems.

The comparative performance of various models for our disease detection process is summarized in Table 1. Among the models under evaluation, GAN + Autoencoder (ResNet) demonstrated the most optimal effectiveness, achieving an accuracy of 97%, precision of 96%, recall of 98%, and an F1-score of 97%. This approach outperformed other architectures such as InceptionV3, MobileNet, and VGG. The superior performance of the GAN + AE (ResNet) model can be attributed to the use of GANs for providing synthetic diseased leaf pictures, which effectively addresses data imbalance.

The performance our model is graphically represented in Fig. 3. The autoencoder's anomaly detection capability enhances the identification of diseased leaves. While other models also performed well, they lacked the robustness and adaptability provided by the GAN + AE framework. This analysis underscores the advantage of combining GANs and autoencoders in plant disease classification, demonstrating their potential for highly accurate and scalable detection systems.

The primary assessment metrics for the GAN + Autoencoder (GAN & AE) model in the procedure for detecting tomato leaf diseases are shown in Fig. 4. The model achieves an impressive 97% accurate, 96% precise, 98% recall and 97% F1-score. These results validate the model's robustness, demonstrating an optimal balance between precision and recall. The model delivers outstanding proficiency across all measures, ensuring reliability in practical applications. Overall, this visualization highlights the exceptional capability of the GAN & AE framework over alternative models.

Table 1 Evaluation measures for the model

Model	Accuracy (%)	Precision	Recall	F1-score
InceptionV3	91.5	0.88	0.89	0.83
MobileNet	88.5	0.84	0.87	0.81
VGG	84	0.79	0.89	0.82
ResNet	97	0.96	0.98	0.97

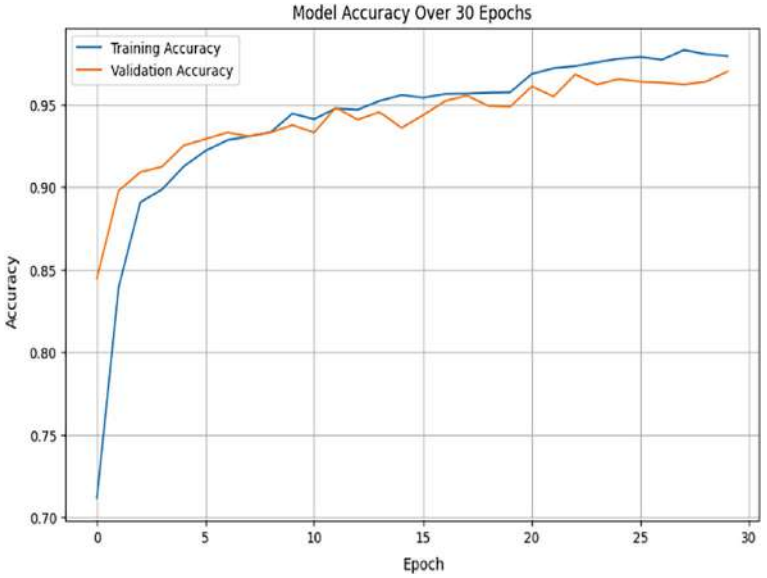


Fig. 3 Model accuracy over 30 Epochs

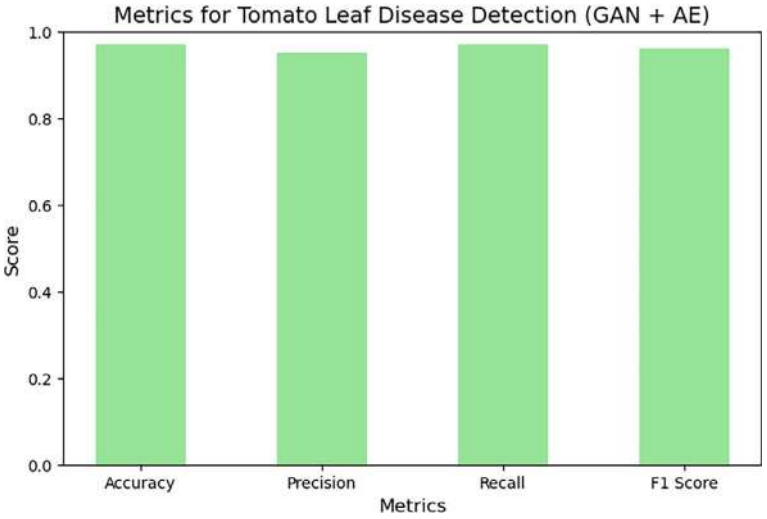


Fig. 4 Performance indicators for the GAN + Autoencoder model for detecting tomato leaf disease

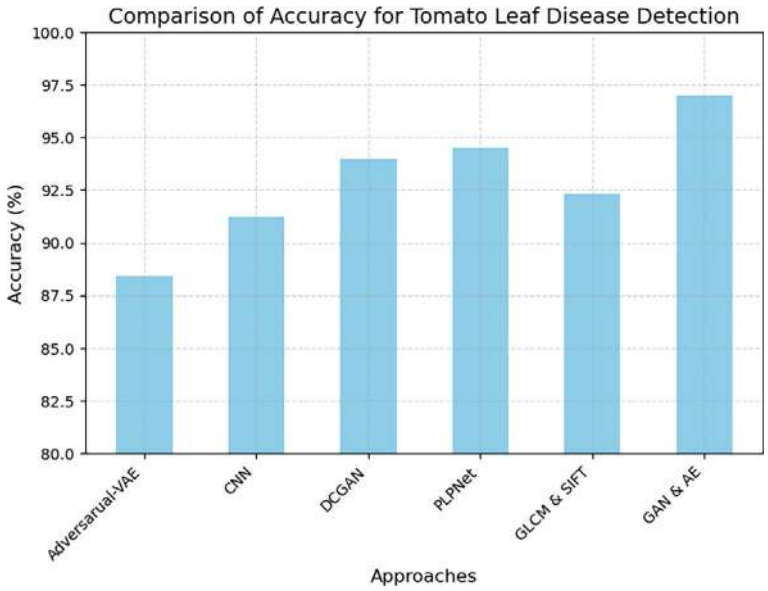


Fig. 5 Accuracy comparison of tomato leaf disease detection models

The levels of accuracy for various samples applied in the diagnosis of diseases of tomato leaves are exemplified in Fig. 5. The models included are Adversarial-VAE [5], CNN [7], DCGAN [4], PLPNet [8], GLCM & SIFT [11], and GAN & AE. Among these, the GAN & AE approach attained the highest accuracy of 97%, showcasing its superior performance. Other models, such as PLPNet and DCGAN, also performed well, achieving accuracies above 90%. This comparison underscores the effectiveness of GAN-based models in enhancing accuracy for plant disease detection tasks.

The confusion matrix presented in Fig. 6 visually illustrates the classification results of the plant disease detection model. The attributes on the diagonal specify correctly predicted cases, and off-diagonal values specify misclassifications. Most of the predictions align with actual labels, with minimal errors observed. The color variations highlight the frequency of classifications, with lighter shades representing higher values.

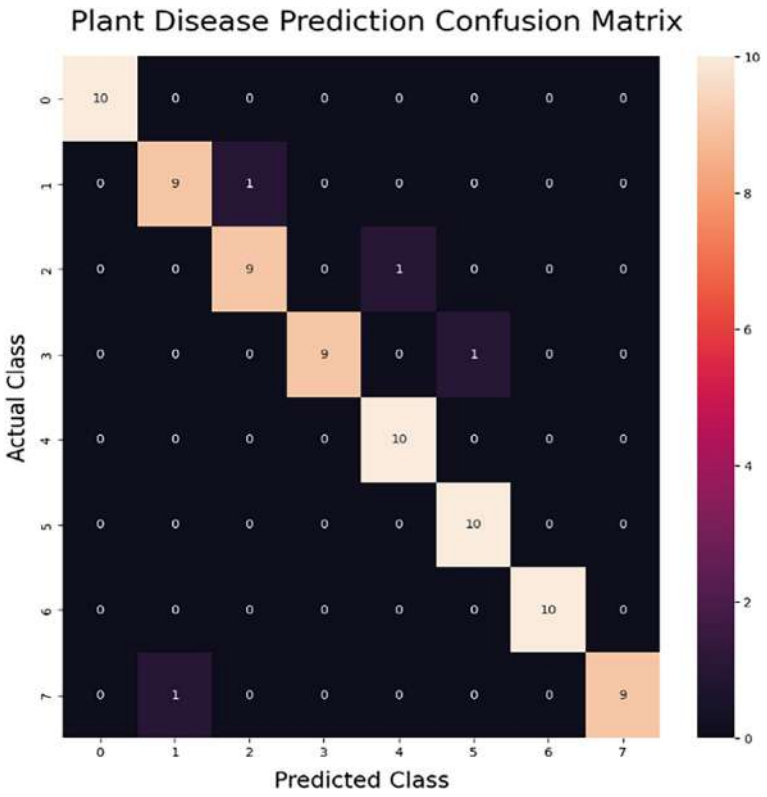


Fig. 6 Disease prediction confusion matrix

5 Conclusion

This research introduces a hybrid deep learning framework that merges Generative Adversarial Networks (GANs) and autoencoders to achieve efficient and accurate tomato leaf disease detection. The model addresses critical limitations found in existing methods, such as poor performance with limited or non-diverse datasets and the challenges of manual inspection. By generating synthetic images and extracting essential features, the proposed method significantly enhances classification accuracy, dataset diversity, and model generalization across various environmental conditions.

Our approach offers an economical, scalable, as well as reliable remedy suitable for both smallholder as well as large-scale farmers, especially in resource constrained settings. It enables real-time monitoring through integration with mobile and drone technologies, enabling farmers to take prompt actions, limit pesticide usage, supports better crop health as well as productivity.

This work advances the field by demonstrating how a hybrid deep learning framework can overcome data scarcity and feature extraction challenges in agricultural disease detection. As a potential direction for future work, the model may be expanded to cover other crops and integrated with IoT devices for a fully automated smart farming solution, further strengthening sustainable agricultural practices.

References

1. Mo, H., Wei, L.: Tomato yellow leaf curl virus detection based on cross-domain shared attention and enhanced BiFPN. Eco. Inform. **85**, 102912 (2025). https://doi.org/10.1016/j.ecoinf.2024.102912
2. Kaushik, M., Prakash, P., Ajay, R., Veni, S.: Tomato leaf disease detection using convolutional neural network with data augmentation. In: 2020 5th International Conference on Communication and Electronics Systems (ICCES), pp. 1125–1132. IEEE, COIMBATORE, India (2020). https://doi.org/10.1109/ICCES48766.2020.9138030
3. Ashok, S., Kishore, G., Rajesh, V., Suchitra, S., Sophia, S.G.G., Pavithra, B.: Tomato leaf disease detection using deep learning techniques. In: 2020 5th International Conference on Communication and Electronics Systems (ICCES), pp. 979–983. IEEE, Coimbatore, India (2020). https://doi.org/10.1109/ICCES48766.2020.9137986
4. Wu, Q., Chen, Y., Meng, J.: DCGAN-based data augmentation for tomato leaf disease identification. IEEE Access **8**, 98716–98728 (2020). https://doi.org/10.1109/ACCESS.2020.2997001
5. Wu, Y., Xu, L.: Image generation of tomato leaf disease identification based on adversarial-VAE. Agriculture **11**(10), 981 (2021). https://doi.org/10.3390/agriculture11100981
6. Basavaiah, J., Arlene Anthony, A.: Tomato leaf disease classification using multiple feature extraction techniques. Wireless Pers. Commun. **115**(1), 633–651 (2020). https://doi.org/10.1007/s11277-02007590-x
7. Sakkarvarthi, G., Sathianesan, G.W., Murugan, V.S., Reddy, A.J., Jayagopal, P., Elsisi, M.: Detection and classification of tomato crop disease using convolutional neural network. Electronics **11**(21), 3618 (2022). https://doi.org/10.3390/electronics11213618
8. Tang, Z., et al.: A precise image-based tomato leaf disease detection approach using PLPNet. Plant Phenomics **5**, 0042 (2023). https://doi.org/10.34133/plantphenomics.0042
9. Abbas, A., Jain, S., Gour, M., Vankudothu, S.: Tomato plant disease detection using transfer learning with C-GAN synthetic images. Comput. Electron. Agric. **187**, 106279 (2021). https://doi.org/10.1016/j.compag.2021.106279.x
10. Agarwal, M., Singh, A., Arjaria, S., Sinha, A., Gupta, S.: ToLeD: tomato leaf disease detection using convolution neural network. Procedia Comput. Sci. **167**, 293–301 (2020). https://doi.org/10.1016/j.procs.2020.03.225
11. Khan, R., Ud Din, N., Zaman, A., Huang, B.: Automated tomato leaf disease detection using image processing: an SVM-based approach with GLCM and SIFT features. J. Eng. **2024**(1), 9918296 (2024). https://doi.org/10.1155/2024/9918296
12. Nagamani, H.S., Sarojadevi, H.: Tomato leaf disease detection using deep learning techniques. IJACSA **13**(1) (2022). https://doi.org/10.14569/IJACSA.2022.0130138
13. Ahmad, I., Hamid, M., Yousaf, S., Shah, S.T., Ahmad, M.O.: Optimizing pre-trained convolutional neural networks for tomato leaf disease detection. Complexity **2020**, 1–6 (2020). https://doi.org/10.1155/2020/8812019
14. Chen, H.-C., et al.: AlexNet convolutional neural network for disease detection and classification of tomato leaf. Electronics **11**(6), 951 (2022). https://doi.org/10.3390/electronics11060951

15. Harakannanavar, S.S., Rudagi, J.M., Puranikmath, V.I., Siddiqua, A., Pramodhini, R.: Plant leaf disease detection using computer vision and machine learning algorithms. Glob. Transit. Proc. **3**(1), 305–310 (2022). https://doi.org/10.1016/j.gltp.2022.03.016

Deep Learning-Driven Detection of Guava Diseases for Smart Agriculture

M. Prasanna Kumari, G. N. V. G. Sirisha, and R. Amith Varma

Abstract Guava (Psidium guajava), a key fruit crop, is susceptible to various diseases that threaten its productivity. This study presents a deep learning and computer vision to detect guava diseases in both leaves and fruits. A Convolutional Neural Network (CNN) was initially implemented, followed by data augmentation and transfer learning to enhance classification performance. Several pre-trained models, including EfficientNetB3, InceptionV3, Xception, VGG16, MobileNetV2, ResNet50V2, DenseNet121 and GoogleNet (Inception V1), were trained and evaluated on six disease categories: Anthracnose, Stylar-End-Rot, Scab, Red-Rust, Phytophthora and disease-free samples. To assess model performance accuracy, precision, recall and F1-score were calculated for every model. While multiple models achieved high accuracy (97–98%), GoogleNet (InceptionV1) outperformed others with 99.16% accuracy. The study's findings demonstrate improvements over previous methods, highlighting the effectiveness of the proposed approach.

Keywords Guava (Psidium guajava) · Deep learning · CNN and transfer learning · GoogleNet (InceptionV1)

1 Introduction

Guava (Psidium guajava) is a widely cultivated fruit in tropical and subtropical regions, contributing to global food security and economic growth. Guava, a member of the myrtle family (Myrtaceae), originates from Mexico, Central America, and northern South America. The Myrtaceae family includes around 150 species of trees and shrubs [1]. However, Guava fruit production is impacted by various diseases

M. Prasanna Kumari (✉) · G. N. V. G. Sirisha · R. Amith Varma
Departmentment of Computer Science and Engineering, SRKR Engineering College Bhimavaram, Bhimavaram, Andhra Pradesh, India
e-mail: m.prasannakumari1234@gmail.com

G. N. V. G. Sirisha
e-mail: sirishagadiraju@srkrec.ac.in

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_22

such as Anthracnose, Stylar-End-Rot, Scab, Red-Rust, Phytophthora can lead to substantial economic losses if not detected and managed promptly. Farmers have good understanding of these diseases, but they often lack awareness of early prevention strategies. As a result, guava production suffers significant loss in guava field. Traditional methods for disease diagnosis are labor- intensive, subjective and prone to error, necessitating the development of automated solutions [2].

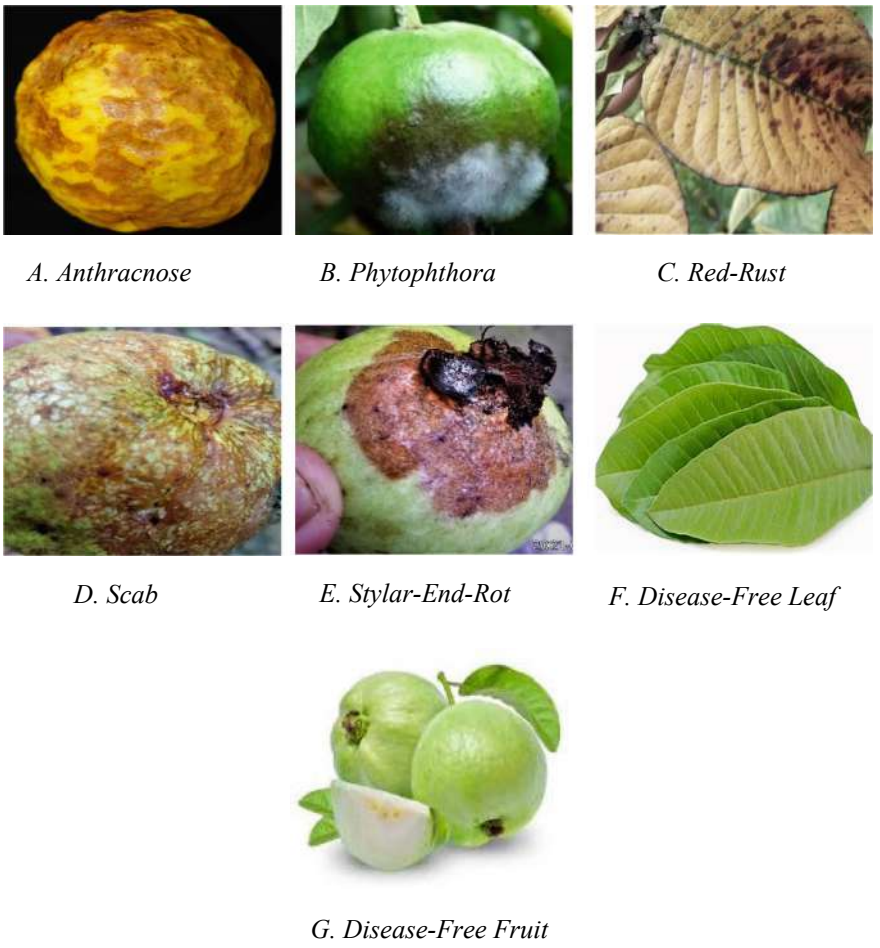
Advancements in deep learning have revolutionized agricultural practices by enabling the rapid and accurate classification of plant diseases. This study identifies the application of deep learning techniques [3–5], particularly Convolutional Neural Networks (CNNs) [6–8] and transfer learning to detect guava diseases from images of leaves and fruits.

1.1 Dataset

In this work, dataset was obtained from agricultural fields in Kalidindi and Bhimavaram, A.P. State, India along with additionally, data was obtained from Mendeley [9] (https://data.mendeley.com/datasets/x84p2g3k6z/1). The images of guava fruits and leaves in the dataset categorized into seven classes Anthracnose, Phytophthora, Red-Rust, Scab, Stylar-End-Rot, Disease-Free Leaf and Disease-Free Fruit.

Figure 1 depicts the dataset sample images (A) Anthracnose, (B) Phytophthora, (C) Red-Rust, (D) Scab, (E) Stylar-End-Rot, (F) Disease-free leaves, (G) Disease-free fruits. Guava production is badly impacted by these diseases and creating great loss to the farmers.

From Table 1 disease-wise data distribution in the dataset. It shows the collection of 6106 labelled images of guava leaves and fruits. Out of 6106 images, 661 are original and 5445 are augmented images. Among them, images for Anthracnose and Disease-Free Fruit are collected from the fields by us and performed data augmentation. Additionally, we include 30 images in Phytophthora, 10 images in Red Rust and 10 images in Disease-Free Leaf which are collected from the fields by us. Totally, out of 661 original images, we collected 184 images of guava leaves and fruits from the fields in Kalidindi and Bhimavaram, A.P. state-India, while 477 images were taken from Mendeley dataset. Nandi, Rabindra Nath, et al. [9] use the Mendeley dataset and achieved 97% accuracy with the GoogleNet model. By adding extra features like Anthracnose and Disease-Free Fruit to the Mendeley dataset we achieved 99% accuracy with GoogleNet model. With this we can detect the diseases of guava more efficiently to get healthier guava crops minimized yield losses, and increased productivity.



A. Anthracnose *B. Phytophthora* *C. Red-Rust*

D. Scab *E. Stylar-End-Rot* *F. Disease-Free Leaf*

G. Disease-Free Fruit

Fig. 1 Dataset sample images

Table 1 Disease-wise data distribution

Disease name	Original images	Augmented images	Total No. of images
Anthracnose	87	335	422
Phytophthora	114	942	1056
Red-Rust	87	1154	1241
Scab	106	864	970
Stylar-end-rot	94	1063	1157
Disease-free-leaf	126	876	1002
Disease-free-fruit	47	211	258

2 Related Work

Misra [10] provides an in-depth look at major guava diseases such as Guava Wilt, Anthracnose, Cercospora Leaf Spot, Guava Scab, and Red Rust, discussing their symptoms, causative agents, and control methods. It suggests the use of fungicides, including Copper Oxychloride and Zineb, to manage fungal infections and highlights the importance of adequate spacing to reduce humidity and prevent disease spread. Additionally, bioagents are recommended as a proactive disease control strategy.

Thilagavathi et al. [11] presents how image processing techniques can be used to detect guava leaf diseases. He used segmentation, color models, and SIFT for feature extraction, and applies SVM and k-NN for classification. Among the tested methods, SVM showed higher accuracy, demonstrating the effectiveness of image processing in plant disease detection.

Almutiry et al. [12] proposes a new framework for accurately classifying multiple types of guava diseases. He used advanced image processing and machine learning techniques to detect and categorize diseases from guava leaf and fruit images. The model improves classification performance and supports early disease diagnosis for better crop management.

Almadhor et al. [13] focuses on identifying guava diseases like canker, mummification, dot, and rust using machine learning on high-quality DSLR images. He used image enhancement and CNN-based classification to detect color and texture changes, aiding in accurate disease detection and better management in guava farming.

Rajbongshi et al. [14] presents a guava image dataset containing both fruit and leaf samples, categorized into six classes. Fruit images include Scab, Stylar End Rot, Phytophthora, and healthy fruit, while leaf images are labeled as either healthy or affected by Red Rust. The dataset the author gives in this paper is helpful for researchers in machine learning and computer vision to build guava disease detection systems that can help to farmers in better crop management.

Tewari et al. [15] explores how deep learning, especially CNNs, can help in solving agricultural issues by focusing on guava disease detection. It targets diseases like Scab, Red Rust, Phytophthora, and Stylar End Rot, which are often hard to detect with traditional methods. By applying CNNs with transfer learning, results high accuracy, with DenseNet169 reaching 99.62%. Even with a small dataset, several models performed exceptionally well, that shows deep learning helps in improving crop health and productivity.

Pathmanaban et al. [16] explores how image processing and computer vision can help detect guava fruit diseases early. It uses a dataset with both digital and thermal images, covering healthy, damaged, and diseased fruits—such as those affected by wilt, Anthracnose, canker, and rot—captured at various maturity stages and drop heights. By identifying issues early, the goal is to support sustainable farming, reduce crop damage, and improve harvest quality.

Mustak Un Nobi et al. [17] introduces a real-time guava leaf disease detection system using a lightweight deep learning model based on MobileNet. That approach

gives fast and accurate classification of diseases while being efficient enough for use on mobile and low-power devices, making it suitable for practical agricultural applications.

Rashid et al. [18] presents a hybrid deep learning system for real-time detection of multiple guava leaf diseases. It integrates GIP-MU-Net for segmentation, GLSM for leaf health classification, and GMLDD with YOLOv5 for disease identification. Using two custom datasets, the model achieved high accuracy and supports better disease management, highlighting the need for further research on environmental and species-related variations.

Mumtaz et al. [19] introduces SidNet, a 33-layer CNN model for detecting guava leaf blight disease. It uses YCbCr color space and data augmentation to handle limited data and incorporates feature extraction inspired by DarkNet-53 and AlexNet. Feature selection is optimized using Binary Gray Wolf Optimization. The model showed high accuracy, achieving up to 99.2% with SVM classifiers, showing strong potential for early disease detection and improved crop yield.

Doutoum et al. [20] addresses key challenges in identifying guava diseases like Canker, Dot, Mummification, and Rust, especially in tropical regions. It uses 1834 leaf images across five categories and tests four CNN models—ResNet50, Inception V3, EfficientNet-B3, and VGG-16. EfficientNet-B3 achieved the highest accuracy at 94.93%. The model enables on-device detection through smartphones, offering a practical tool for farmers. However, this study notes that data quality and variety is still impact performance.

3 Proposed Work

The system being developed leverages DL models, such as CNNs and pre-trained models like InceptionV3, Xception, VGG16, MobileNetV2, ResNet50V2, DenseNet121, GoogleNet (InceptionV1), to automate the detection of guava diseases. By incorporating data augmentation techniques, it increases the model's precision in identifying diseases like Anthracnose, Red-Rust, Stylar-End-Rot, Scab, Phytophthora. The system offers a scalable and economical method aims to support early disease detection, helping to mitigate crop losses and boost agricultural productivity.

Figure 2, shows the proposed architecture for generating the trained Deep learning model. The collected data will be annotated with class labels. Then it will be pre-processed by resizing images followed by augmentation. Before being used for model training, the data split into validation, training and test sets. In that basic model CNN (i.e., 2 Convolutional layers which extracts image features, 2 Max Pooling layers which reduces image dimensions to enhance computational efficiency, 1 Flatten layer converts the multi-dimensional input into a 1D vector to connect convolutional layers to dense layers, 2 Fully Connected layers which supports classification) and the pre-trained models (InceptionV3, MobileNetV2, Xception, VGG16, DenseNet121,

ResNet50V2 and GoogleNet (InceptionV1)) will be applied and it gives to the best trained model.

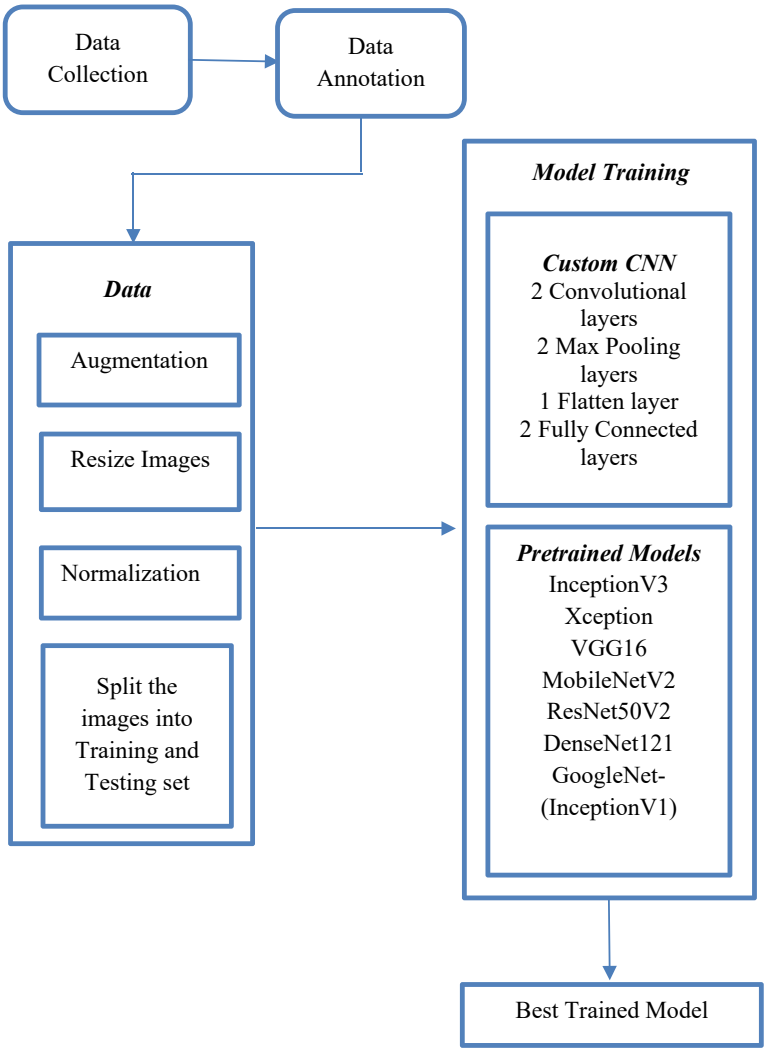


Fig. 2 Proposed system architecture

3.1 Data Preprocessing

In the guava disease detection, data preprocessing involves several crucial methods to make sure that input data meets requirements for model training, clean and well-structured.

3.1.1 Collecting and Labelling the Data

Images of guava leaves and fruits affected by diseases like Anthracnose, Red-Rust, Scab, Stylar-End-Rot, Phytophthora, including healthy samples are gathered for analysis.

3.1.2 Data Cleaning

Any noise or irrelevant data, such as images with poor quality or unrelated content, is removed to improve the dataset's quality [21].

3.1.3 Resizing and Scaling the Images

To meet the input criteria, the images are scaled to a uniform 150*150-pixel size of the CNN models. Pixel values are normalized, typically scaled between 0 and 1, for more efficient processing by the neural network.

3.1.4 Data Augmentation

To prevent overfitting and enhance model robustness, augmentation techniques such as random rotation, flipping, zooming, shifting and brightness adjustments are used to make the dataset more diverse and larger than it actually is [22].

3.1.5 Splitting the Dataset

The dataset is separated into validation, training and test sets so that the model may be tested and trained on different data segments to determine its effectiveness.

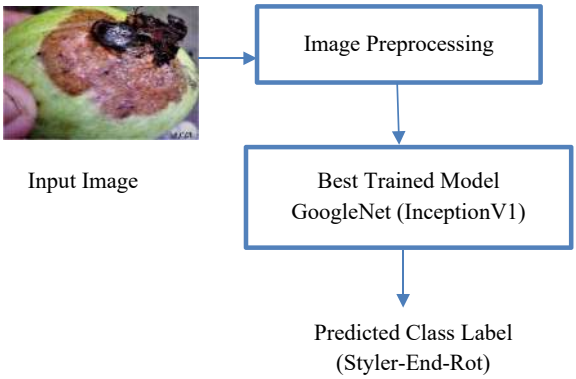


Fig. 3 Model inferencing

3.1.6 Label Encoding

The target classes Red-Rust, Anthracnose, Scab, Phytophthora, Stylar-End-Rot, Healthy are transformed into numerical format, typically one-hot encoding is used for multi-class classification.

This Process ensures that the data is prepared effectively for optimal performance during the training phase.

3.2 Model Inferencing

Figure 3 shows an image of a guava fruit is first loaded and prepared to meet the model's input specifications, which involves resizing and normalizing the image. Once the preprocessing is complete, the image is input into the model, which generates a prediction regarding the most probable disease category associated with the image, enabling a precise classification i.e. stylar-end-rot. Such an approach is essential in agriculture for the timely identification and management of guava diseases, ultimately contributing to reduced crop losses and enhanced yields.

4 Result Analysis

4.1 Comparison Table

The baseline model used in this study was **CNN**. This model achieved a test accuracy 84% with a loss of 0.5198. In order to improve the performance transfer learning is used.

Table 2 Comparison of training accuracies and validation accuracies using different pretrained networks

Model	Train_Acc	Val_Acc	Train Loss	Val_Loss
CNN	0.95	0.84	0.15	0.51
InceptionV3	0.87	0.91	0.31	0.21
Xception	0.92	0.93	0.22	0.15
VGG16	0.92	0.94	0.24	0.15
MobileNetV2	0.95	0.97	0.10	0.07
ResNet50V2	0.95	0.97	0.13	0.10
DenseNet121	0.96	0.98	0.08	0.05
GoogleNet (InceptionV1)	0.99	0.99	0.03	0.02

From Table 2 it can be observed that training accuracy, validation accuracy, training loss, validation loss of CNN model and pre-trained models such as Xception, DenseNet121, VGG16, MobileNetV2, InceptionV3, ResNet50V2, GoogleNet (InceptionV1) are compared.

Among that GoogleNet (InceptionV1) get highest accuracy. GoogleNet is a deep learning model introduced by Google in 2014, known for its efficient Inception modules that process multiple filter sizes simultaneously. It features factorized convolutions to reduce computational complexity while maintaining high accuracy. The model includes auxiliary classifiers to improve gradient flow, aiding faster convergence during training. With 22 layers, it outperforms earlier architectures like AlexNet and VGG16 while using fewer parameters. GoogleNet won the ILSVRC 2014 competition and remains widely used for image classification tasks. In guava disease detection, it achieved 99% accuracy.

Figure 4 shows the validation accuracies according to the Table 2 of different models of guava disease detection in a graph form. Compare to the other models GoogleNet model stands out with the highest bar in the histogram represents that it is the best model in guava disease detection.

4.2 Evaluation Metrics

From Table 3 indicates the performance comparison of various deep learning models for guava disease detection using accuracy, precision, recall and F1-score. It shows that:

1. Baseline CNN Performance—The CNN model achieves 84% accuracy, indicating moderate performance without transfer learning.
2. Impact of Transfer Learning—Pretrained models significantly improve classification accuracy, with InceptionV3 reaching 91%, Xception and VGG16 at 94%, MobileNetV2 and ResNet50V2 at 97%.

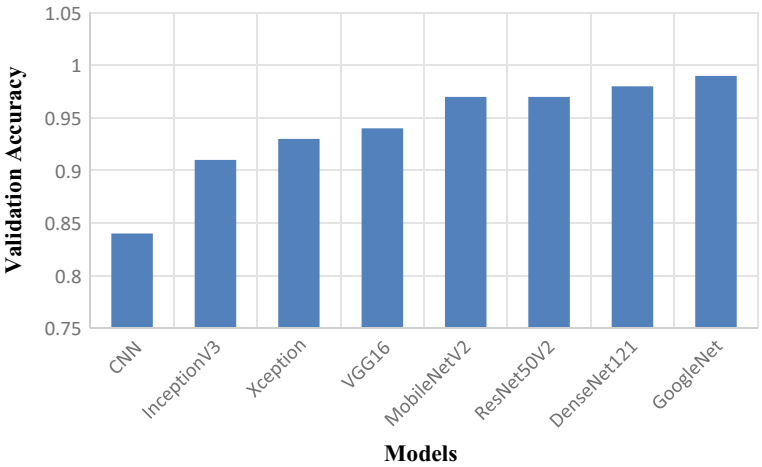


Fig. 4 Validation accuracies of different models for guava disease detection

Table 3 Performance metrics of different pretrained CNN models

Model	Accuracy	Precision	Recall	F1-score
CNN	0.84	0.84	0.84	0.84
InceptionV3	0.91	0.90	0.90	0.90
Xception	0.94	0.93	0.93	0.93
VGG16	0.94	0.94	0.94	0.94
MobileNetV2	0.97	0.97	0.97	0.97
ResNet50V2	0.97	0.97	0.97	0.97
DenseNet121	0.98	0.98	0.98	0.98
GoogleNet (InceptionV1)	0.99	0.99	0.99	0.99

3. Top Performing Models—DenseNet121 98% and GoogleNet 99% provide the best results, with GoogleNet achieving the highest accuracy and reliability.
4. Overall Trend—As model complexity and depth increase, performance improves, highlighting the effectiveness of advanced architectures in guava disease detection.

This analysis suggests that GoogleNet (InceptionV1) is the best choice for accurate classification.

5 Conclusion

In conclusion, to detect diseases of guava leaves and fruits by using traditional methods it takes a lot of human labour and it is very difficult to differentiate between disease types due to their similar shape, texture and colour. But this work successfully developed a DL-based system to detect and classify the guava diseases, specifically targeting Anthracnose, Red-Rust, Stylar-End-Rot Scab, Phytophthora. The System applied basic CNN and pre-trained models like MobileNetV2, Xception, DenseNet121, VGG16, InceptionV3, ResNet50V2, and GoogleNet (InceptionV1). DenseNet121 and GoogleNet (InceptionV1) are able to accurately predict the guava diseases. The performances of these models are 98% and 99% respectively. Techniques for data augmentation were applied to reduce overfitting and enhance the model's capacity for generalization. This work highlights the effectiveness of transfer learning in agricultural applications and provides a practical tool for farmers and experts to diagnose guava diseases efficiently. Evaluation measures like validation accuracy, F1-score, recall, and precision were used to assess the model's efficiency, with the top-performing model delivering strong results in detecting guava diseases. This automated system offers a valuable solution for farmers by facilitating early disease detection and enhancing crop management practices, leading to healthier guava crops, minimized yield losses, and increased productivity.

Future work—Developing a mobile or web-based application for real-time disease detection to assist farmers.

References

1. Dheir, I., Abu-Naser, S.S.: Knowledge based system for diagnosing guava problems. Int. J. Acad. Inf. Syst. Res. (IJAISR) **3**(3), 9–15 (2019)
2. Simbeye, D.S.: Iot Based Early Identification of Guava (Psidium Guajava) Leaves and Fruits Diseases. Available at SSRN 4270295
3. Mostafa, A.M., et al.: Guava disease detection using deep convolutional neural networks: a case study of guava plants. Appl. Sci. **12**(1), 239 (2021)
4. Chug, A., et al.: A novel framework for image-based plant disease detection using hybrid deep learning approach. Soft. Comput. **27**(18), 13613–13638 (2023)
5. Kadam, V., Patil, M.P., Rishabh Kumar, M.: Mobile Application for Predicting Diseases with Providing Remedies on Guava Plant Leaves with the Help of Deep Learning Techniques and Cloud Computing
6. Asim, M., et al.: Varietal discrimination of guava (Psidium guajava) leaves using multi features analysis. Int. J. Food Prop. **26**(1), 179–196 (2023)
7. Foo, C.F., et al.: Android-based app guava leaf diseases ıdentification using convolution neural network. J. Adv. Res. Appl. Sci. Eng. Technol. 73–88 (2024)
8. KILCI, O., KOKLU, M.: Classification of Guava Diseases Using Features Extracted from Squeezenet with Adaboost and Gradient Boosting (2024)
9. Nandi, R.N., et al.: Device-friendly Guava fruit and leaf disease detection using deep learning. International Conference on Machine Intelligence and Emerging Technologies. Springer Nature Switzerland, Cham (2022)

10. Misra, A.K.: Guava diseases—their symptoms, causes and management. Diseases of Fruits and Vegetables: Volume II: Diagnosis and Management, pp. 81–119. Springer Netherlands, Dordrecht (2004)
11. Thilagavathi, M., Abirami, S.: Application of image processing in diagnosing guava leaf diseases. Int. J. Sci. Res. Manage. (IJSRM) **5**(7), 5927–5933 (2017)
12. Almutiry, O., et al.: A novel framework for multi-classification of guava disease. Comput., Mater. Continua **69**(2), 1915–1926 (2021)
13. Almadhor, A., et al.: AI-driven framework for recognition of guava plant diseases through machine learning from DSLR camera sensor based high resolution imagery. Sensors **21**(11), 3830 (2021)
14. Rajbongshi, A., et al.: A comprehensive guava leaves and fruits dataset for guava disease recognition. Data Brief **42**, 108174 (2022)
15. Tewari, V., Azeem, N.A., Sharma, S.: Automatic guava disease detection using different deep learning approaches. Multimedia Tools Appl. **83**(4), 9973–9996 (2024)
16. Pathmanaban, P., Gnanavel, B.K., Anandan, S.S.: Comprehensive guava fruit data set: digital and thermal images for analysis and classification. Data Brief **50**, 109486 (2023)
17. Mustak Un Nobi, Md., et al.: GLD-DET: guava leaf disease detection in real-time using lightweight deep learning approach based on MobileNet. Agronomy **13**(9), 2240 (2023)
18. Rashid, J., et al.: Real-time multiple Guava leaf disease detection from a single leaf using hybrid deep learning technique. Comput., Mater. Continua **74**(1), 1235–1257 (2023)
19. Mumtaz, S., et al.: A hybrid framework for detection and analysis of leaf blight using guava leaves imaging. Agriculture **13**(3), 667 (2023)
20. Doutoum, A., Eryiğit, R., Tuğrul, B.: Classification of guava leaf disease using deep learning. World Sci. Eng. Acad. Soc. (WSEAS) **20**(38) (2023)
21. Mirjat, R.M., et al.: A framework for guava wilt disease segmentation using k-means clustering and neural network techniques. VAWKUM Trans. Comput. Sci. **12**(1), 76–93 (2024)
22. Haq, Z.A., Jaffery, Z.A., Mehfuz, S.: Precision harvesting: comparative analysis of machine learning and generative AI-based classifiers for guava fruit maturity assessment using thermal imaging. CyTA-J. Food **22**(1), 2401588 (2024)

Machine Learning Approach for Fraud Detection in Banking Data

M. Sai Lakshmi Sarvani, D. Rajani, and K. Rohan Reddy

Abstract Credit card fraud has become a critical issue with the rise of digital transactions, necessitating advanced detection mechanisms. This project presents a Django-based fraud detection system leveraging machine learning and deep learning models, including Support Vector Machines, Naive Bayes, Logistic Regression, Decision Trees, and XGBoost, optimized for high accuracy. The system processes transaction data in real time, utilizing SMOTE for class imbalance handling and structured databases for efficient data management. With an interactive web interface, it provides fraud detection visualizations through pie charts, bar charts, and splines while allowing service providers to download prediction datasets and monitor performance. XGBoost emerged as the most effective model, achieving 99.9% accuracy, demonstrating the system's scalability and reliability in real-world fraud prevention.

Keywords Credit card fraud detection · Machine learning · Django web application · Data preprocessing · Fraud prediction

1 Introduction

Credit card fraud has always been a major problem during the web going era, where the internet's usage of mobile phones and comparable items occurred on a daily basis. Paying by credit card has gone over the top leading to the increase of the same in fraudulent activities as a result, people have faced big financial losses both as individuals and as financial institutions. Normally, fraud detection methods, that is to say, sieves of traditional nature, are hardly competent enough to handle the growing complexity and massive number of transactions; so they are not entirely helpful in stopping the fraudsters' clever tactics.

M. Sai Lakshmi Sarvani (✉) · D. Rajani · K. Rohan Reddy
Computer Science and Engineering, Institute of Aeronautical Engineering (JNTUH), Hyderabad, India
e-mail: sarvanimotukuru@gmail.com

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_23

The advent of machine learning (ML) and deep learning (DL) is the beginning of a new era and their significant contribution to fraud detection with their high accuracy and efficiency has been realized with the passing of the project. This project's main aim is to develop a solid credit card fraud detection system by using the most advanced ML and DL algorithms. The system, by data mining, wants to find the misleading patterns in the owner's past financial activities, enabling the real-time unearthing and ceasing of the fraudulent activities.

The system is built as a web application based on Django that practices a group of machine learning models consisting of a diverse set of algorithms such as Naive Bayes, Logistic Regression, Decision Trees, and XGBoost all of which are polished through hyperparameter tuning. The backend has a structured database that takes care of transaction and user data management, and the front end provides an intuitive interface for service providers to visualize results, view detection rates, and download datasets.

In addition to this, the algorithm also uses the most advanced pre-processing method called the feature scaling, Synthetic Minority Oversampling Technique (SMOTE) (for class imbalance), and feature engineering which are intended to prepare good training data. In comparison of various models, with the XGBoost algorithm, the results show that the best performance comes out of this one which scores over 99.9% accuracy. Such projects prove the efficacy of fraud detection by modern algorithms as well as the importance of their practical implementation by offering a scalable and effective solution to credit card fraud mitigation in real-world scenarios.

2 Related Work

2.1 Literature Review

A large amount of studies in the material are concerned with the evaluation of financial fraud through statistical methods. Namely, it was identified that the most important studies were the ones that resorted to the use of ordinary least squares (OLS) regression and autoregressive (AR) models for the assessment of financial fraud. According to J. Khaksar, the research develops several regression models and explores the association between fraud and auditor characteristics in going concern situations in emerging economies. To be more specific, the authors deliver information on how the finding reliability can be raised [1]. The A. Cordis, the study presents the exploration of the effect of political alignment on corporate fraud convictions, which provides unique insights into the connection between politics and fraud. The authors leverage public data from 2003 through 2018 of parties' US affiliation and related corporate fraud convictions at the state level [2]. The study is solved by the use of OLS technique to analyse financial factors of financial fraud, which is the base of fraud triangle [3].

Abakarim et al. [4] study was designed to provide a credit card fraud detection model that works in real-time and uses deep learning approaches most efficiently. The suggested method focused on increasing the detection rate with minimizing the false alarm. According to the results, deep learning could be used as an imposing tool for credit card companies to deal with rapid changes in various fraud methods. This paper compared dominating rough set approaches with providing an automated machine learning algorithm auto loan fraud. These results indicated that rough set processes are more efficient in extracting useful aims thereby improving accuracy, thereby, highlighting the significance of the cutting-edge selection criteria [4]. Arora et al. [5] application of artificial intelligence to solve the limited credit card transaction case was the first step by Arora and co-researchers. It proposed a way to test the reduction of false positives, and the results showed an increase in detection rates, which is why it can be easily applied in practice [6].

Blaszczyński et al. [7] paper compared rough set techniques based on dominance and classic machine learning methods for fraud detection in the automobile loan sector. Their report revealed that a rough set algorithm was the most efficient as it was successful in extracting the most pertinent features and thus improving prediction accuracies, hence indicating the need for advanced feature selection strategies [5]. Branco et al. [8] scholars proposed one-level recurrent neural networks (RNNs) that are interleaved with respect to fraud detection. A model's ability in processing the time series data through simultaneous interweaving has been point out in the research. This was done by making necessary changes in the sequence data; hence, the systems could differentiate fraudulent activities among various of them [7].

Fang et al. [9] researched credit card fraud detection based on machine learning techniques, highlighting the problems of datasets with an imbalanced class. It was observed that the potential of ensemble learning techniques for fraud detection is very high in this particular area. Furthermore, it was also found that there is an enhancement of detection performance with a decrease in the rate of misclassification [10]. Forough and Momtazi [11] work introduced an ensemble of LSTM models for fraud detection, where the best features of the various deep learning models are combining through ensemble methods. However, they sounded the alarm for the uptick in the number of fake transactions on the internet, particularly in identify fraud instances within imbalanced data [12].

Makki et al. [13] article by Makki and co-authors was an experimental study on imbalanced classification methods for the detection of credit card fraud. The research stressed out that the applicability of resampling techniques and hybrid models was in the effective addressing of the class imbalance problem, thus the retrieval of high precision and recall scores [8]. Matloob et al. [14] survey represents a healthcare fraud detection method based on predicting and mining the sequences of the data. Though focusing on another area, the method suggested in this research could be applied to identify fraud in transactional datasets [15].

Benchaji et al. [16] study proposed a credit card fraud detection model that makes use of long short-term memory (LSTM) recurrent neural networks. The authors of the project have emphasized the ability of the LSTM model to process sequential data to achieve higher detection rates than those traditional machine learning approaches

[17]. Hu et al. [18] LightGBM with asymmetric error control for credit card fraud detection, that which Hu and colleagues investigated is the one that was demonstrated to be successful by the use of one conceptual model. The model was able to predict the number of frauds while minimizing the false alarms, and thus costs were kept at a low level, which developers [19].

Kim et al. [20] presented a hierarchical cluster-based deep neural network approach to fraud detection of job placement data. The research was not at all limited to the domain of credit card transactions being, still, it was seen that deep clustering systems could be effectively applied when dealing with several intricate forms of fraudulent typing [16]. Kim and Kim [21] scientists here designed a neural classifier model based on a fraud density map which could locate the potential areas where fraud might exist in the transaction dataset and thus the system was better at discerning which were cases of fraud and eliminating the cases of false negatives [9]. Kousika et al. [22] found that the practicing of their method caused a significant reduction in incidence of e-payment fraud. They preferred to employ their knowledge in problems like feature selection and data balancing methods rather than simply replacing the feature which habitually has led to overtraining issues [11].

2.2 Existing System

Inferior credit card fraud detection mechanisms mainly utilize rule-based ways that detect fraud through predefined rules or conditions. Indeed, transactions that are greater than a specific amount or are located in very rare places could be labeled as suspected if they are suspicious. This type of system is often supported by simple statistical models or historic data analysis to recognize irregularities. Also, manual investigations are recurrently performed to confirm reported transactions, which makes the whole process lengthy and resource-consuming.

Although these measures have been successful in preventing clear fraud, they hardly handle the increased sophistication of illegal practices. While transaction volumes are increasing, regular procedures are getting less and less effective in both storing and pursuing data related to tendencies in fraud. Furthermore, many basic articles in this area talk about the total percentage of frauds being detected and do not refer to particular fraudulent transactions. This weak point prevents the widespread application of such systems [23–27].

2.3 Disadvantages of Existing System

- The system has not implemented Classification on Imbalanced Data. The system does not explicitly address class imbalance, where fraudulent transactions are

much fewer than legitimate ones. Without techniques like SMOTE or cost-sensitive learning, the model may favor the majority class, leading to undetected fraud cases (false negatives) and reduced detection accuracy.

- There are many systems, including mentioned in base papers, that just give a percentage of frauds detected, without specifying which actual transactions are found fraudulent. This feature limits the usability of these kinds of systems for real-time fraud prevention.

3 Proposed Work

3.1 Proposed System

The proposed system introduces a robust credit card fraud detection mechanism that leverages advanced machine learning (ML) and deep learning (DL) algorithms to overcome the limitations of traditional approaches. This system is implemented as a Django-based web application, integrating powerful models such as Naive Bayes, Logistic Regression, Decision Trees, and XGBoost. The models are trained and fine-tuned to ensure high accuracy and scalability, enabling real-time prediction of fraudulent transactions. The backend is designed to store and manage transaction data securely, while the frontend provides an intuitive interface for fraud analysis and visualization.

Unlike existing systems that focus only on the percentage of fraud detection, the proposed system identifies specific fraudulent transactions. By processing detailed transaction data, the system pinpoints anomalies and classifies them as fraud or genuine, giving stakeholders actionable insights. Advanced data preprocessing techniques, including feature scaling and handling class imbalances with Synthetic Minority Oversampling Technique (SMOTE), ensure the models are trained on balanced, high-quality datasets. This approach minimizes false positives and negatives, enhancing the reliability of the predictions.

The system also emphasizes user interaction and interpretability through various visualization tools. Service providers can access fraud detection ratios, download prediction datasets, and view model performance metrics in the form of interactive charts, such as pie, bar, and line graphs. Additionally, the integration of state-of-the-art algorithms, such as XGBoost and deep learning architectures, enables the system to dynamically adapt to evolving fraud patterns. The result is a scalable, efficient, and proactive fraud detection solution designed to meet real-world challenges.

3.2 Advantages

The proposed system delivers significant improvements in accuracy, scalability, and usability compared to traditional methods. Its use of advanced algorithms ensures accurate detection, reducing both false positives and false negatives. By pinpointing fraudulent transactions, the system provides practical, transaction-level insights that traditional approaches and base papers often lack. The ability to retrain and adapt the models with new data further enhances its effectiveness against emerging fraud techniques.

The web application interface, built with Django, offers a user-friendly experience for service providers. Features such as interactive visualizations, downloadable datasets, and real-time detection enable stakeholders to monitor, analyze, and act on fraud cases seamlessly.

3.3 Activity Diagram

An activity diagram which represents the energetic flow of the fraud detection system, illustrating how the process goes step by step from **data collection** to the final result. It starts off with the accrual of transaction data which is further **processed** through cleaning, normalization, and, first preparation of the dataset. Having implemented processing, the system moves on to the **data analysis** step and further **selects features** the most relevant to training the model.

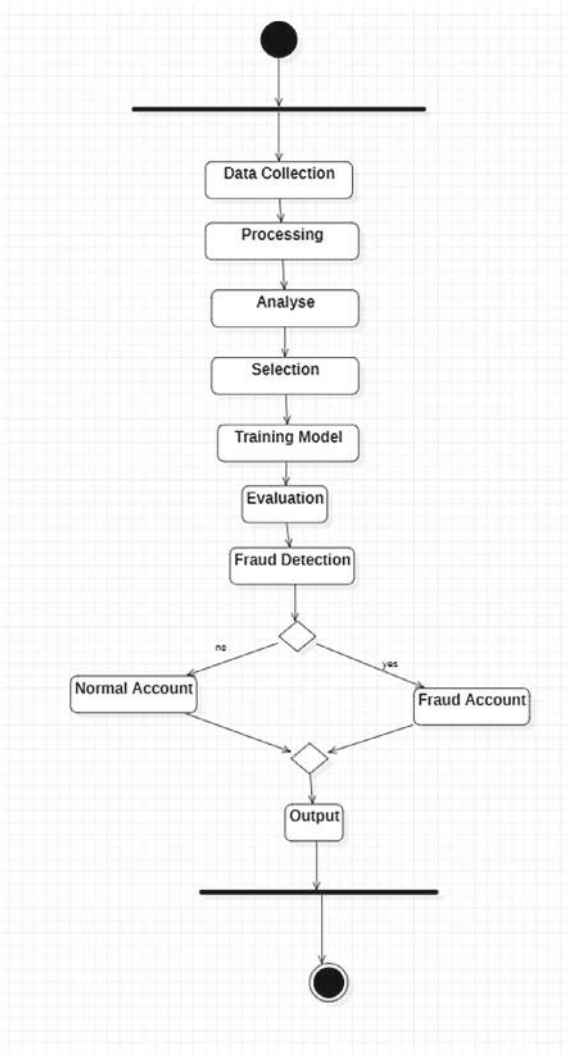
Following feature selection, the **model training** stage is initiated, where machine learning models are created and optimized. These models are then executed and their performance is gauged using markers that are suitable. After the model has gone through the **evaluation** process, the system enters the fraud detection step, where transactions are sorted according to the known patterns.

If a transaction is tagged, then it is classified as either a **Normal Account** or a **Fraud Account**. In conclusion, the system produces an output that gives a clear result, which is then further processed, shown, or saved, in the users through the application interface (Fig. 1).

3.4 System Architecture

Proposed architecture of the system reduces the complexity of credit card fraud detection by the implementation of functions such as data processing and user interaction. The service provider permits users to log in, train, and test datasets, and view the accuracy of test datasets in bar charts, predict fraud detection rates, and finally, download the predicted datasets (Fig. 2).

Fig. 1 Activity diagram



A web server responds to the inquiries of users and enables the communication with the web database that stores and retrieves datasets, results, and user data. That is to say, these are the things happening for the quickly mean of rushing data to the time of the end of the process.

For Remote Users, the system will not only provide the functions of registering, log in, fraud type prediction, and personalized profile access, but it will also include the following tweet server which would possibly give a chance of a real-time or social media communication besides the main unit of interaction about the system and its design.

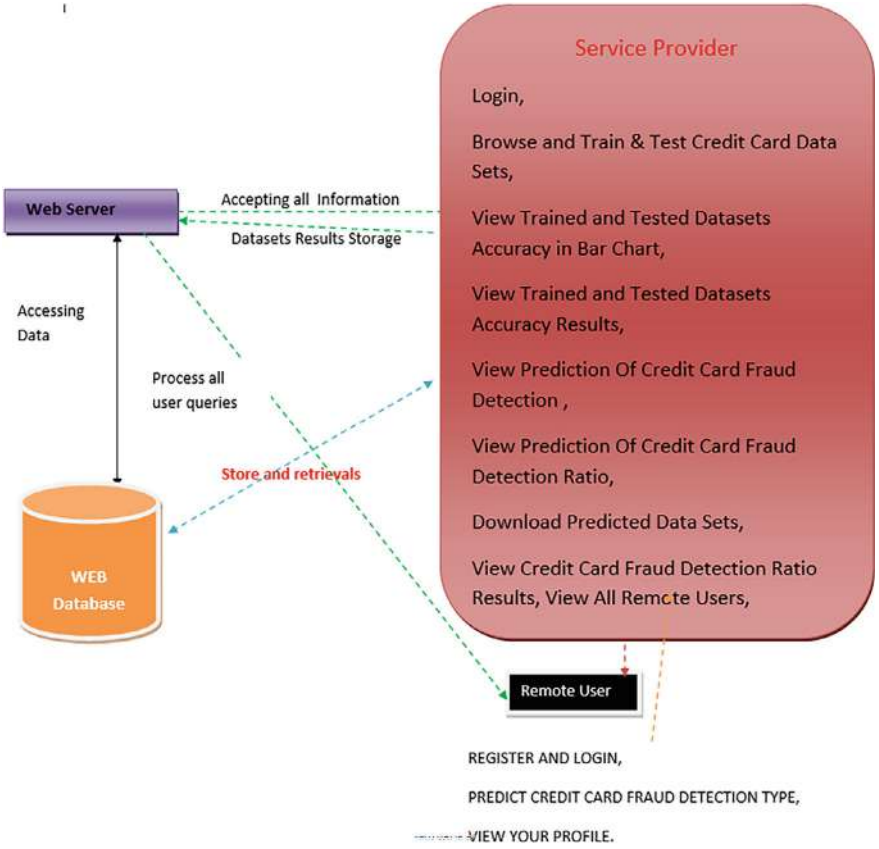


Fig. 2 System architecture

3.5 Flow Chart

3.5.1 Remote User

The process begins when the remote user attempts to log in to the system. The system first prompts the user to input their username and password (Fig. 3).

Step 1: Login

- The user enters their username and password to gain access.

Step 2: Status Check

- The system verifies the credentials entered. This is represented by the status decision point in the flowchart.
- If the credentials are correct, the user proceeds to the next steps of the process.

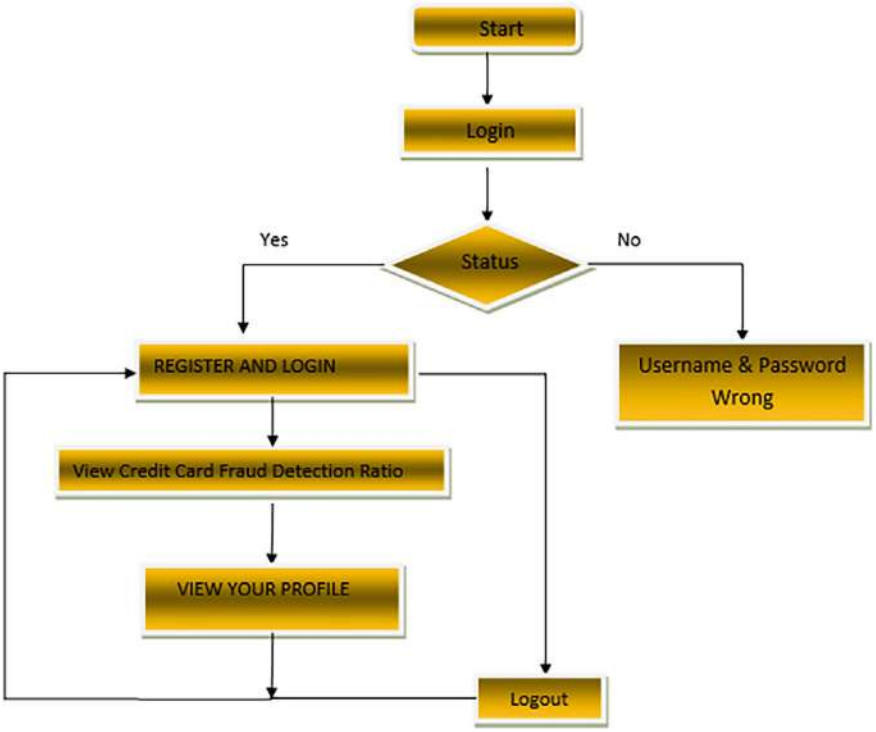


Fig. 3 Overview of remote user

- If the credentials are incorrect, the system will display an error message: "Username & Password Wrong", and the user is prompted to re-enter their login information.

Step 3: Registration (If Needed)

- If the user has not yet registered, they are prompted to register and log in. Once registered, they can proceed with the system's features.

Step 4: View Credit Card Fraud Detection Ratio

- After a successful login, the user is directed to a screen where they can view the credit card fraud detection ratio. This feature likely provides statistics or data on the detection of fraudulent activities related to credit cards, enhancing the user's ability to monitor fraud levels.

Step 5: View Profile

- Next, the user can view their profile. This section might display the user's personal data, settings, and preferences. This is an essential feature for users to manage and update their details in the system.

Step 6: Logout

- After interacting with the system, the user has the option to log out of the application. This step ensures that the session is securely ended, preventing unauthorized access.

3.5.2 Service Provider

The process begins with the service provider logging into the system. The system checks the provided credentials, and the process follows the steps based on the login status.

Step 1: Login

- The service provider is prompted to enter their username and password for authentication.

Step 2: Status Check

- The system verifies the entered credentials at the status decision point.
- If the credentials are correct, the provider is allowed to proceed with the following tasks.
- If the credentials are incorrect, the system displays an error message: "Username & Password Wrong", and the provider is prompted to re-enter the correct credentials.

Step 3: Browse and Train/Test Credit Card Data Sets

- Upon successful login, the service provider can browse and test credit card data sets. This includes the ability to view and manipulate the data sets to train and test their models for fraud detection.

Step 4: View Trained and Tested Datasets Accuracy

- After training the data, the provider can view the accuracy of the datasets.
- Accuracy in a Bar Chart: The provider can visualize the results in a bar chart format for better clarity.
- Accuracy Results: Detailed results of the trained and tested datasets are also available, allowing for deeper insights into the model's performance (Fig. 4).

Step 5: View Predictions of Credit Card Fraud Detection

- The service provider can then view predictions regarding credit card fraud detection. This feature likely shows the probability of fraud based on the trained models.

Step 6: View Prediction of Credit Card Fraud Detection Ratio

- A more specific analysis, the fraud detection ratio, is available to show the proportion of accurate fraud predictions in relation to total predictions made.

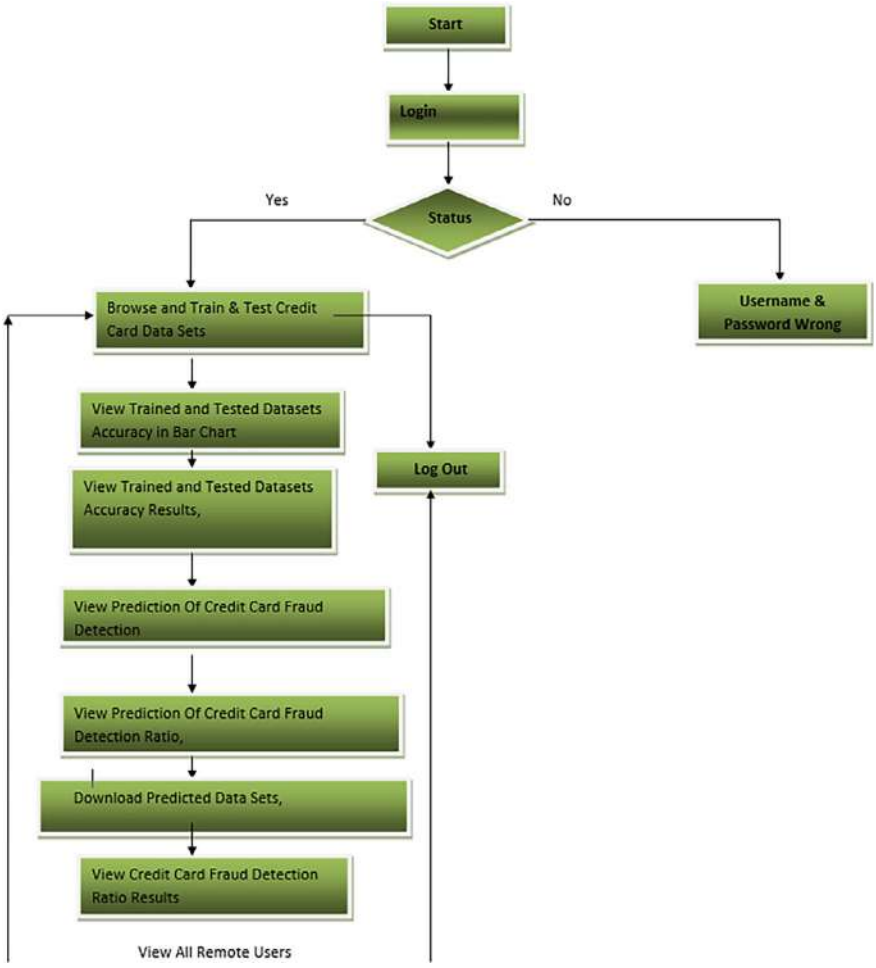


Fig. 4 Overview of service provider

Step 7: Download Predicted Data Sets

- Once satisfied with the predictions and results, the service provider has the option to download the predicted data sets for further analysis or integration into other systems.

Step 8: View Credit Card Fraud Detection Ratio Results

- Finally, the provider can view the credit card fraud detection ratio results, which offers a summary of how well the model has performed overall.

Step 9: Log Out

- After completing the tasks, the service provider can log out of the system, ending their session securely.

3.6 Django Framework

Django is a web framework powered by Python for high-level web development that allows fast, reliable, and scalable web applications. It also employs the use of reusable components, adheres to the model-view-controller (MVC) architectural pattern, and comes with many built-in features—such as authentication, database management, and URL routing—that are super useful because it enables the creation of complex applications. As for credit card fraudulent detection, Django can be utilized to develop the backend of a system that integrates machine learning models to predict fraudulent transactions. Through the use of Django's ORM to manage transaction data and the combination of a machine learning model (for instance, decision trees, neural networks) provided in Python libraries like Scikit-learn or TensorFlow, developers can finally set up a fraud detection system that operates in real time by scrutinizing transaction patters. The analysis of web interface results may help users visualize detection findings, enabling them to oversee, modify, and also obtain good system response.

3.7 Logistic Regression (LR) Algorithm

Logistic regression analysis studies the relationship between a categorical dependent variable and a set of independent (explanatory) variables. The term logistic regression is used when the dependent variable has only two values, for example, 0 and 1 or Yes and No. The term multinomial logistic regression is typically used for the situation when the dependent variable has three or more distinct values, such as Married, Single, Divorced, or Widowed. The practical use of the procedure is similar, even though the type of data used for the dependent variable is different from that of a multiple regression.

Logistic regression is a technique that is at odds with discriminant analysis for the classification of categorical-response variables. There are many statisticians who consider logistic regression to be more flexible and therefore better for most modeling situations than discriminant analysis. This is because logistic regression assumes independent variables are normally distributed, as discriminant analysis does not.

This software will solve binary logistic regression and multinomial logistic regression on the numeric and categorical independent variables. It shows the regression equation along with the goodness of fit, odds ratios, confidence limits, likelihood, and deviance. It does the whole process of residual analysis, where it includes diagnostic residual reporting and plotting. It has the independent variable subset selection search

capabilities that will look for the best regression model with the fewest independent variables. It delivers the prediction intervals and ROC curves for better classification. It also lets you validate your results by automatically classifying those rows that are not used during the analysis.

4 Results

The highest accuracy at 99.94% was reached by Logistic Regression, which proves its outstanding performance for the detection of fraudulent transactions. With an accuracy of 99.93%, Support Vector Machine (SVM) almost reached the performance level of the LR model and is not far below. Both Decision Tree Classifier and Gradient Boosting Classifier achieved 99.92% accuracy, which is good example of their efficiency and reliability in detection.

Figure 5, This figure displays the input interface where users can manually enter transaction attributes for fraud prediction. Features such as Time, V1 to V28, and Transaction Amount must be entered based on the anonymized credit card dataset.

The Fig. 6, presents the system's final prediction output after the user inputs transaction values. Once the "Predict" button is clicked, the system evaluates the data using the trained machine learning model and returns the prediction result. The above example shows a "**Non-Fraud Transaction**", meaning the input values are legitimate.

The Fig. 7, displays the Admin Portal of the fraud detection system. Through this interface, administrators can manage and monitor datasets, view fraud detection ratio results, and access visual analytics like pie charts and line graphs. The portal offers options to upload and train datasets, evaluate model accuracies, download predicted

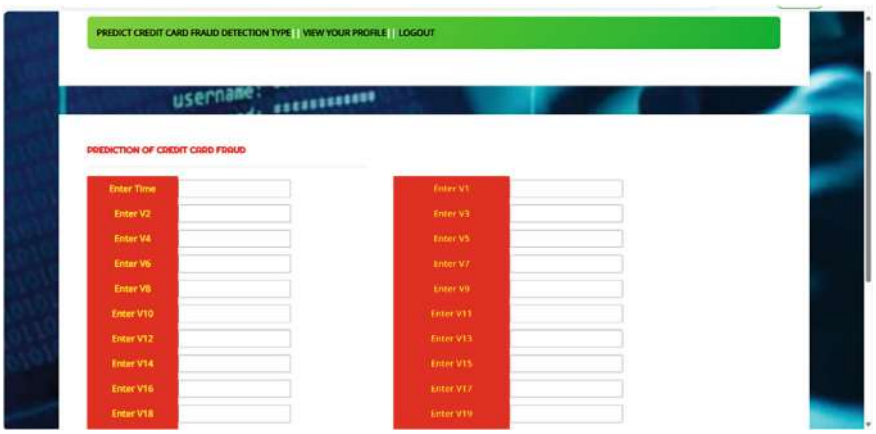


Fig. 5 User interface for manual transaction entry

Fig. 6 Prediction result output

results, and manage remote users. It serves as the backend control center, ensuring smooth operation, dataset handling, and system performance monitoring.

Although the Logistic Regression and Gradient Boosting Classifier models were the best performers, they succeeded in solving the obvious problem of dataset imbalance as these models are able to learn very complex data structures and thus, they are capable to spot even the rarest cases of fraud. The algorithm works in such a way that it eliminates false positives and false negatives, making it possible for it to be practically applied in the real world. The comparative analysis of the models confirms the power of the system and its capacity to be utilized in the finance field, particularly in the banks' prudential assessments. By this means, the system then

Fig. 7 Snapshot of admin portal

becomes an essential weapon in fighting financial fraud and increasing transaction security (Figs. 8, 9 and 10).

The Credit Card Fraud Detection Ratio shows that 75% of the transactions are non-fraudulent, while 25% are fraudulent. This imbalance is typical in fraud detection systems, where the majority of transactions are legitimate. Despite the smaller proportion of fraud, detecting the 25% of fraudulent transactions is crucial to prevent financial losses. The challenge lies in ensuring that the system accurately identifies these fraudulent cases without being biased toward the larger, non-fraudulent class. High detection accuracy for fraud is essential for the system's effectiveness and reliability.

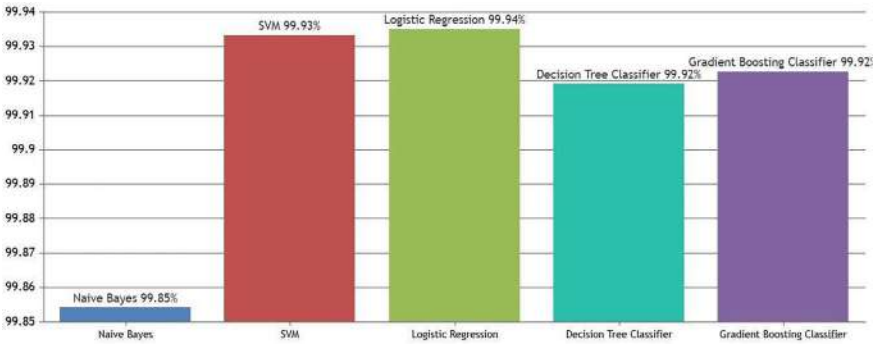


Fig. 8 Trained and tested datasets accuracy (bar graph)

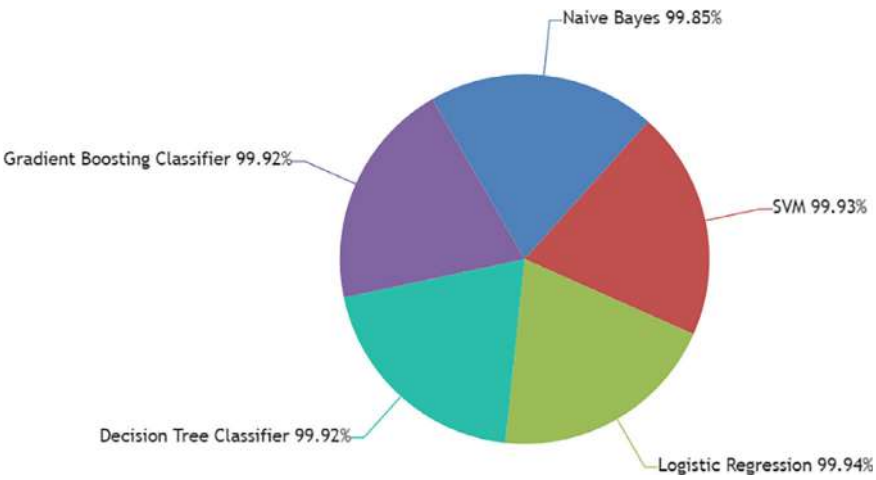


Fig. 9 Trained and tested datasets accuracy (pie chart)

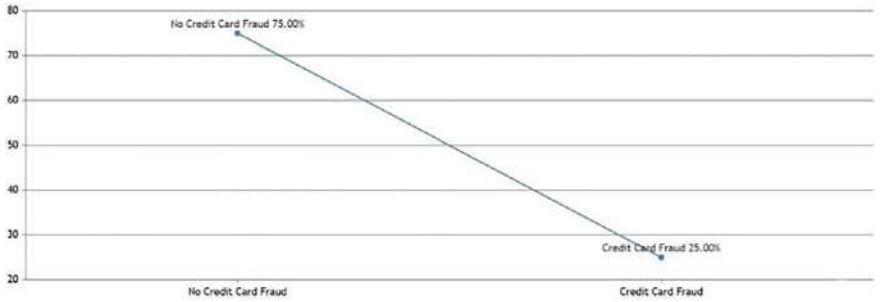


Fig. 10 Credit card fraud detection ratio

4.1 Scalability and Performance

Scalability is a must for a credit card fraud detection system, as the amount of transactions it can handle should go up without a decrease in BOSMAX. The system becomes bloated with the growing amount of transaction data, and it has to utilize a method of processing larger datasets efficiently, dealing with a larger number of users, or using new features like advanced fraud detection algorithms or real-time transaction monitoring. Methods such as distributed computing, cloud infrastructure, and parallel processing have the capacity for scaling, thus enabling the system to meet demand and be effective without the need for additional delays.

Performance denotes the capability of the system to discover the fake transactions correctly in addition to reducing the two wrong types of results. The main factor is the fraudulent detection system operating with efficiency, and real-time output is available even when the environment is crowded. In order to guarantee the high performance, the algorithms should be sure to use models that are light for prompt fraud detection and/or employ methods for the reduction of data dimensionality in massive datasets. Constant tuning and retraining of the models based on the most recent transaction statistics are also required to let the models adapt to new fraud tactics, thus, assuring accuracy and reliability in fraud detection.

4.2 Ethical Concerns

Data Privacy is a major main ethical concern in the field of fraud detection, because they are obliged to deal with extremely sensitive personal and financial information. The most important thing in user data is that it is stored securely, encrypted, and anonymized where applicable in accordance with legal requirements such as GDPR. This is to make sure the personal and financial details remain only with the users and are not shared with unauthorized parties thus it maintains user privacy and increases the trust in the system. Moreover, companies ought to ask for strict control and thus be sure that only those they assign are able to deal with sensitive data.

Algorithmic Transparency is, in this case, a different ethical question; since most of the fraud detection programs employ machine learning models operating as "black boxes." This makes the problem a matter of explanation and understanding rather than a simple decision to approve/disapprove. Moreover, clear communication to customers, explaining that wrong decisions can sometimes happen, is also very important. Transparency in the decision-making process establishes trust between the system and the user which later on helps the user to present the case of the disputed transaction further.

Fairness and Bias are vital ethical components of the process, primarily when the non-uniform distribution of fraud detection models by the demographic struck is one of the central areas to be concerned about. If the available training data to train these models is biased, the result will be unfair, typically, the transactions from different groups will be verified in different ways so there is a strong chance of discrimination to one of them. It is necessary to employ different, diverse data sets and regularly check models for bias to prevent this situation. The action of creating fairness in fraud detection can be well described as the method of taking care of both the vulnerable side and fairness of the system, which will increase public confidence in its reliability and integrity.

5 Conclusion

The computer determines a method of fraud recognition by using high-end machine learning and deep learning. The system has achieved an accuracy of more than 99.9% with the models Logistic Regression, SVM, and Gradient Boosting, thus showing that it is able to detect fraudulent transactions with a very high precision. The system becomes more practical by finding specific fraudulent transactions and not just giving fraud detection percentages; in this way, it is flexible and can be utilized to prevent fraud in real-time. In addition, the incorporation of data pre-processing techniques, e.g. SMOTE for handling class imbalances, leads to such consistent and unbiased training, that it tampers both false positives and negatives to the minimum.

Besides technical aspects, the system is developed in such a way that its Django-based interface and interactive visualization tools make it available for both the service providers and stakeholders. The features that offer real-time fraud predictions, detailed fraud detection data, and downloadable datasets are basically the ones that account for comprehensive fraud management. Its modular design and capability to be scalable make it very easy to connect with existing financial platforms, which in turn offers a proactive, efficient, and reliable credit card fraud solution. The project is a great example of the use of machine learning, deep learning as well as modern web technologies and their combination, which results in scalable and effective real-life problem solutions.

References

1. Khaksar, J., Salehi, M., Lari DashtBayaz, M.: The relationship between auditor characteristics and fraud detection. J. Facil. Manage. **20**(1), 79–101 (2022). https://doi.org/10.1108/jfm-02-2021-0024
2. Cordis, A.: Political alignment and corporate fraud: evidence from the United States of America. J. Appl. Acc. Res. (2023). https://doi.org/10.1108/jaar-06-2022-0159
3. Rahman, M.J., Jie, X.: Fraud detection using fraud triangle theory: evidence from China. J. Financ. Crime **31**(1), 101–118 (2024). https://doi.org/10.1108/jfc-09-2022-0219
4. Abakarim, Y., Lahby, M., Attioui, A.: An efficient real time model for credit card fraud detection based on deep learning. In: Proceeding 12th International Conference Intelligence Systems: Theories Applied, pp. 1–7 (2018). https://doi.org/10.1145/3289402.3289530
5. Arora, V., Leekha, R.S., Lee, K., Kataria, A.: Facilitating user authorization from imbalanced data logs of credit cards using artificial intelligence. Mobile Inf. Syst. **2020**, 1–13 (2020). https://doi.org/10.1155/2020/8885269
6. Abdi, H., Williams, L.J.: Principal component analysis Wiley Inter-discipl. Rev., Comput. Statist. **2**(4), 433–459 (2010). https://doi.org/10.1002/wics.101
7. Balogun, A.O., Basri, S., Abdulkadir, S.J., Hashim, A.S.: Performance analysis of feature selection methods in software defect prediction: a search method approach. Appl. Sci. **9**(13), 2764 (2019). https://doi.org/10.3390/app9132764
8. Branco, B., Abreu, P., Gomes, A.S., Almeida, M.S.C., Ascensão, J.T., Bizarro, P.: Interleaved sequence RNNs for fraud detection. In: Proceeding 26th ACM SIGKDD International Conference Knowledge Discovery Data Mining, pp. 3101–3109 (2020). https://doi.org/10.1145/3394486.3403361
9. Fang, Y., Zhang, Y., Huang, C.: Credit card fraud detection based on machine learning. Comput., Mater. Continua **61**(1), 185–195 (2019). https://doi.org/10.32604/cmc.2019.06144
10. Bandaranayake, B.: Fraud and corruption control at education system level: a case study of the Victorian department of education and early childhood development in Australia. J. Cases Educ. Leadersh. **17**(4), 34–53 (2014). https://doi.org/10.1177/1555458914549669
11. Forough, J., Momtazi, S.: Ensemble of deep sequential models for credit card fraud detection. Appl. Soft Comput. **99**, 106883 (2021). https://doi.org/10.1016/j.asoc.2020.106883
12. Błaszczyński, J., de Almeida Filho, A.T., Matuszyk, A., Szeląg, M., Słowiński, R.: Auto loan fraud detection using dominance-based rough set approach versus machine learning methods. Expert Syst. Appl. **163**, 113740 (2021). https://doi.org/10.1016/j.eswa.2020.113740
13. Makki, S., Assaghir, Z., Taher, Y., Haque, R., Hacid, M.-S., Zeineddine, H.: An experimental study with imbalanced classification approaches for credit card fraud detection. IEEE Access **7**, 93010–93022 (2019). https://doi.org/10.1109/ACCESS.2019.2927266
14. Matloob, I., Khan, S.A., Rahman, H.U.: Sequence mining and prediction-based healthcare fraud detection methodology. IEEE Access **8**, 143256–143273 (2020). https://doi.org/10.1109/ACCESS.2020.3013962
15. Cartella, F., Anunciacao, O., Funabiki, Y., Yamaguchi, D., Akishita, T., Elshocht, O.: Adversarial Attacks for Tabular Data: Application to Fraud Detection and Imbalanced Data (2021). arXiv:2101.08030
16. Benchaji, I., Douzi, S., Ouahidi, B.E.: Credit card fraud detection model based on LSTM recurrent neural networks. J. Adv. Inf. Technol. **12**(2), 113–118 (2021). https://doi.org/10.12720/jait.12.2.113-118
17. Lad, S.S., Adamuthe, A.C.: Malware classification with improved convolutional neural network model. Int. J. Comput. Netw. Inf. Secur. **12**(6), 30–43 (2021). https://doi.org/10.5815/ijcnis.2020.06.03
18. Hu, X., Chen, H., Zhang, R.: Short paper: credit card fraud detection using LightGBM with asymmetric error control. In: Proceeding 2nd International Conference Artificial Intelligence for Industries (AII), pp. 91–94 (2019). https://doi.org/10.1109/AI4I46381.2019.00030
19. Dornadula, V.N., Geetha, S.: Credit card fraud detection using machine learning algorithms. Proc. Comput. Sci. **165**, 631–641 (2019). https://doi.org/10.1016/j.procs.2020.01.057

20. Kim, J., Kim, H.-J., Kim, H.: Fraud detection for job placement using hierarchical clusters-based deep neural networks. Int. J. Speech Technol. **49**(8), 2842–2861 (2019). https://doi.org/10.1007/s10489-019-01419-2
21. Kim, M.-J., Kim, T.-S.: A neural classifier with fraud density map for effective credit card fraud detection. In: Yin, H., Allinson, N., Freeman, R., Keane, J., Hubbard, S. (eds.) Intelligent Data Engineering and Automated Learning, vol. 2412, pp. 378–383. Springer, Berlin, Germany (2002). https://doi.org/10.1007/3-540-45675-9_56
22. Kousika, N., Vishali, G., Sunandhana, S., Vijay, M.A.: Machine learning based fraud analysis and detection system. J. Phys., Conf., **1916**(1), 012115 (2021). https://doi.org/10.1088/1742-6596/1916/1/012115
23. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition (2015). arXiv:1512.03385
24. Lima, R.F., Pereira, A.: Feature selection approaches to fraud detection in e-payment systems. In: Bridge, D., Stuckenschmidt, H. (eds.) E-Commerce and Web Technologies, vol. 278, pp. 111–126. Springer (2017). https://doi.org/10.1007/978-3-319-53676-7_9
25. Lucas, Y., Jurgovsky, J.: Credit Card Fraud Detection Using Machine Learning: A Survey (2020). arXiv:2010.06479
26. Zhou, H., Chai, H.-F., Qiu, M.-L.: Fraud detection within bankcard enrollment on mobile device based payment using machine learning. Front. Inf. Technol. Electron. Eng. **19**(12), 1537–1545 (2018). https://doi.org/10.1631/FITEE.1800580
27. Mekterovi¢, I., Karan, M., Pintar, D., Brki¢, L.: Credit card fraud detection in card-not-present transactions: where to invest?. Appl. Sci. **11**(15), 6766 (2021). https://doi.org/10.3390/app11156766

A Communication-Efficient Federated Learning Framework: Reducing Rounds via Adaptive Model Aggregation

Yogita Sachin Narule and Kalpana Sunil Thakre

Abstract This research proposes a communication-efficient federated learning (FL) framework, leveraging adaptive model aggregation to optimize the balance between communication cost and performance of the model. The methodology employs dynamic client selection and aggregation strategies to reduce unnecessary communication while maintaining model accuracy. Clients train local models on decentralized data; only significant updates are transmitted to the central server for aggregation. By adjusting communication intervals and evaluating the quality of client updates, the framework minimizes overhead without compromising model convergence. Experimental results demonstrate a significant reduction in communication rounds, achieving high model accuracy (over 90%) with fewer updates. The framework is scalable and suitable for real-world applications with constrained bandwidth and computational resources. Overall, this approach enhances the efficiency of federated learning by dynamically adjusting communication strategies, ensuring high performance and cost-effectiveness.

Keywords Federated learning · Adaptive aggregation · Communication-efficient · Model convergence · Dynamic client selection · Distributed learning

Y. S. Narule (✉) · K. S. Thakre
Department of Computer Engineering, Marathwada Mitramandal's College of Engineering (Affiliated to Savitribai Phule Pune University, Pune), Karvenagar, Pune, India
e-mail: yogitanarule@gmail.com; narule.yogita@kitcoek.in

K. S. Thakre
e-mail: kalpanathakre@mmcoe.edu.in

Y. S. Narule
Department of Computer Science and Engineering, Kolhapur Institute of Technology's College of Engineering, Kolhapur, India

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_24

1 Introduction

Federated Learning (FL) is a distributed machine learning paradigm that enables collaborative model training across multiple decentralized devices or servers without sharing raw data. By keeping the data confined on devices, a crucial feature in sensitive applications like healthcare, banking, and smart devices, this strategy improves privacy. Only model updates, like weights or gradients, are sent to a central server, which compiles them to update a global model, rather than the actual data [1]. The decentralized training approach in FL reduces the risk of data breaches and complies with data protection regulations, such as GDPR. Moreover, FL can leverage the computational power of edge devices, reducing the need for centralized processing. But FL has a lot of difficulties, mostly because of communication overhead [2]. Because clients and the server need to communicate model updates often, communication costs might create a bottleneck, particularly in large-scale networks. When clients are operating in locations with unreliable network circumstances or have limited resources, this difficulty is magnified.

The communication cost that arises from frequent exchanges between the central server and distributed clients is one of the biggest obstacles in federated learning [3]. Each communication round involves all clients delivering their local model modifications to the server, which combines these updates to develop the global model. To get the desired level of model performance, this process is performed numerous times [4]. Since FL usually functions in situations (such as mobile devices or IoT networks) where bandwidth and communication costs are constrained, the quantity of communication rounds plays a major role in determining the overall performance of the system [5]. The communication cost increases with the number of rounds needed, which harms scalability and real-time deployment in real-world applications [6]. Making FL more efficient requires minimizing the number of rounds while preserving model performance. Furthermore, the problem is made more complex by the heterogeneity of data and processing power among clients, as some may need to contribute more frequently than others. This increases the requirement for communication-efficient solutions.

Current approaches to minimize communication overhead in federated learning mostly concentrate on methods like quantization, model pruning, and gradient compression, which try to minimize the amount of data that is transferred between clients and the server [7, 8]. Although these techniques can lower the cost of communication for each cycle, they sometimes come with trade-offs regarding convergence speed or model fidelity. Asynchronous updates and client selection techniques are further tactics to lessen the requirement for constant connection [9]. These methods could, however, result in discrepancies between the client and global models, which could impede convergence or possibly cause model divergence [10]. Despite these efforts, there is still a lack of information in the literature about methods that adaptively modify the communication frequency in response to the model's learning curve. The majority of techniques rely on set communication intervals, which can be ineffective, particularly in locations with different topographies [11]. A more dynamic

and adaptive model aggregation approach is therefore required, one that minimizes communication rounds without compromising model correctness or convergence time.

To solve these issues, the research proposes a communication-efficient federated Learning Framework that reduces the communication rounds through adaptive client selection and model aggregation. The work is implemented using the FLOWER Framework, a novel, open-source FL framework. The Server and client modules are implemented for Human Activity Recognition using the UCI MHEALTH dataset. The clients train the model locally, and the server uses an adaptive model aggregation technique. It dynamically adjusts the frequency and strategy of communication based on model convergence, client performance, and data distribution. By maximizing the trade-off between communication cost and model performance in federated learning, the suggested architecture improves flexibility. To make sure that communication occurs only when necessary, it adjusts the frequency of communication rounds based on each client's input and the overall global model convergence. Furthermore, the system minimizes unnecessary communication by dynamically adjusting aggregation algorithms to send only significant changes from the most impacting clients. Scalability and effective learning in dispersed environments are guaranteed by the framework, which strikes a balance between lower communication costs and the requirement to retain high model accuracy.

2 Related Work

Communication-Efficient Federated Learning: Federated Learning (FL) presents a major communication efficiency problem because of the large volume of data that needs to be transferred between clients and the central server during training. Numerous approaches have been put out to deal with this problem. Through the use of quantization techniques to limit update precision or sparsification of the gradients, gradient compression approaches seek to minimize the amount of the gradient updates exchanged in each round [12]. By selecting and sending just the most crucial weights or model parameters, model pruning aims to lower communication costs by reducing the amount of model updates. When the precision of model updates is reduced by quantization (e.g., by using fewer bits to indicate weights), the communication bandwidth is dramatically reduced without sacrificing good model performance [13]. Asynchronous communication is an additional technique that minimizes idle times and bandwidth consumption by allowing clients to exchange updates at different times instead of waiting for synchronization at every cycle. Even with these developments, dynamic techniques that strike a compromise between communication frequency and convergence and accuracy of the models are still need to be investigated.

Model Aggregation Techniques: Federated learning is centered around model aggregation. Federated Averaging (FedAvg) is a popular aggregation technique in

which the server averages local model updates from clients to generate a global model [14, 15]. FedAvg is useful in many situations, however, it assumes that client data is identically distributed and independent (IID), which may not hold true in real-world applications where client data can vary greatly. To get around this restriction, FedProx was created [16]. It is an extension of FedAvg that adds a proximal term to handle data heterogeneity and makes sure local models do not drastically differ from the global model. Though both approaches are efficient at gathering information, their communication efficiency in heterogeneous contexts is limited since they do not dynamically adjust to changing conditions among clients and instead assume fixed communication intervals [17]. Other aggregation approaches, such as SCAFFOLD and FedNova, have made tweaks to address drift between client models, but they still rely on frequent communication cycles, which limit scalability.

Adaptive Strategies in Federated Learning: Adaptive techniques have been investigated in federated learning to further enhance model convergence and communication efficiency. Faster convergence with fewer communication rounds is made possible by adaptive learning rates, which dynamically modify each client's learning rate based on variables like data distribution, local model performance, or network conditions [18, 19]. Additionally, dynamic communication intervals have been proposed, in which the quality of client updates or the model's learning progress is used to modify the frequency of communication between clients and the server [20]. As training advances and the model converges, for example, clients may communicate less often in the later phases of training when model updates have less of an influence. In doing so, it lessens pointless communication without sacrificing model performance [21, 22]. To maximize communication efficiency, an adaptive strategy that incorporates client selection and model aggregation procedures has not yet been fully integrated into most adaptive systems, which instead concentrate on specific elements like learning rates or communication intervals [23]. To close that gap, this study introduces an adaptive architecture that dynamically modifies the communication approach as well as the client selection to enhance overall performance.

The study [24] proposes FedDyn, a novel federated learning (FL) method that introduces a dynamic regularization approach for distributed training. The fundamental idea behind FedDyn is to, over communication rounds, dynamically adjust the loss function at each participating client to guarantee that the model converges to a stationary point of the global empirical loss. This dynamic adjustment resolves the discrepancy between the local and global optima caused by data heterogeneity by bringing local device models into line with the global model. FedDyn is shown to outperform traditional FL methods like FedAvg, FedProx, and SCAFFOLD, particularly in reducing communication costs. Across a wide range of datasets, including Shakespeare, CIFAR-10, and MNIST, the technique achieves faster convergence and large communication savings while maintaining strong performance even in non-identically distributed (non-IID) data settings. The approach is well-suited for large-scale and real-world FL applications where communication efficiency is a top requirement because it has been shown to converge at a rate of O(1/T) for both convex and non-convex loss functions. FedDyn assures that local updates are consistent with

the global objective by enabling exact minimization of the local loss at each client. This results in significant reductions in the number of communication rounds needed to reach a target accuracy. This approach is scalable for real-world FL settings and is particularly useful in situations including wide dispersal, heterogeneous data, and unstable communication lines.

The study [25] presents Federated Learning (FL) as an innovative approach to training machine learning models on decentralized data, such as data generated on mobile devices, without sharing the raw data with a central server. The Federated Averaging (FedAvg) technique is presented in this work, and it combines iterative model averaging at a central server with local stochastic gradient descent (SGD) on each client. This approach tackles the main issues with decentralized learning, such as heterogeneous (non-IID) data distribution, restricted connection capacity, and data privacy. The authors show that FedAvg achieves up to a 100 $\times$ decrease in communication rounds and greatly lowers communication expenses when compared to typical distributed SGD. Multiple datasets, such as MNIST, CIFAR-10, and a large language modelling job, were used in the experiments, and the results demonstrate that FedAvg performs well even with non-IID and unbalanced data distributions, which are common in real-world federated learning scenarios. The method is scalable, resilient to data heterogeneity, and communication-efficient, making it appropriate for large-scale applications like IoT and mobile systems.

The study [26] addresses the challenge of reducing communication costs in Federated Learning (FL). FL allows training models on decentralized data located on client devices (e.g., mobile phones), while keeping the data local for privacy reasons. The authors suggest using both sketching updates and structured updates to cut down on uplink traffic. By limiting model updates to a low-rank or sparse format, structured updates lower the transmission volume of data. Conversely, before being sent to the server, sketched updates use methods like subsampling, quantization, and random rotations to compress the model updates. Studies using Reddit datasets (using LSTM for next-word prediction) and CIFAR-10 datasets (using convolutional networks) demonstrate that these methods can cut communication by up to two orders of magnitude with negligible effect on model performance. Because of this, the suggested techniques maintain a high level of model correctness even in situations when communication capacity is limited, like in mobile and Internet of Things networks. These methods are particularly helpful for reducing the amount of communication required during the training of intricate models such as deep neural networks.

In order to enhance the convergence of DFL, the study [27] proposes a novel non-uniform quantization of model parameters. It reduces quantization distortion by adaptively modifying the quantization levels using the Lloyd-Max approach applied to DFL (LM-DFL). The LM-DFL's convergence guarantee is proven independent of the convex loss hypothesis. Based on LM-DFL, a novel doubly adaptive DFL is proposed that takes into account both the increasing number of quantization levels to minimize the quantity of information shared during training and the adjusted quantization levels for non-uniform gradient distributions.

The study [28] addresses key challenges in wearable healthcare, such as data privacy and the need for personalization. The authors propose FedHealth, a federated transfer learning framework that uses homomorphic encryption and federated learning to aggregate health data from multiple organizations while maintaining privacy. FedHealth creates a personalized healthcare model by using transfer learning to refine a global model on the cloud and locally on user devices; the framework shows a 5.3% improvement in activity recognition accuracy over traditional methods. FedHealth is extensible, versatile, and can be applied to various healthcare applications like activity monitoring, cognitive disease detection, and more. FedHealth represents a scalable solution for maintaining data privacy while providing personalized healthcare.

The study [29] explores the performance of federated learning in dynamic environments where data and solutions evolve. To address the main issues with federated learning, the authors present a modified version of the FedAvg algorithm that permits asynchronous operation, partial agent participation, and adapts to non-IID input. They put forth a concept in which local updates are carried out by each agent and then aggregated by the central server. Three major elements that impact performance are identified by the study: a tracking term associated with the learning rate, model variability among agents, and data variability at each agent. The approach operates well in both stationary and non-stationary settings, according to experimental results, with the step size, agent heterogeneity, and model drift influencing performance. The theoretical solutions are validated by the authors through numerous experiments, and they offer assurances of convergence.

The study [30] presents FLOWER, a novel federated learning (FL) framework designed to address the challenges of scalability, system heterogeneity, and seamless integration between simulation and real-world FL settings. With its Virtual Client Engine (VCE), Flower's flexible, language-agnostic, and framework-agnostic implementations allow researchers to run large-scale FL experiments with millions of clients while accounting for heterogeneous edge devices with varying computational resources and network conditions. Flower's key features include its ability to handle both simulated and real-world edge devices, its open-source, extendable nature, and its efficient resource management. Additionally, the framework incorporates secure aggregation protocols for privacy-preserving FL. Flower outperforms other FL frameworks in scalability and system-level heterogeneity, making it a valuable tool for both academic research and industrial-scale FL deployments.

The study [31] addresses the issue of high communication costs in federated learning (FL), which arise due to the need for frequent model updates between clients and the central server. By overlapping the model training phase with the model communication phase, the authors' innovative framework, Overlap-FedAvg, increases communication efficiency. This technique minimizes idle time and boosts overall communication efficiency by enabling model uploads and downloads to happen concurrently with training. To provide steady convergence and address the possibility of gradient staleness resulting from parallelization, Overlap-FedAvg includes a gradient compensation algorithm to further improve performance. To expedite the convergence of the model, Nesterov Accelerated Gradients (NAG) are

also utilized. Further reductions in communication overhead are possible because the framework is interoperable with additional data compression techniques. Comparing Overlap-FedAvg to classic FedAvg, experimental results on image classification and natural language processing tasks show a considerable reduction in communication costs without sacrificing model accuracy.

The approach described in [32] uses deep learning to identify human actions based on sensor data from wearable devices. The technique combines Convolutional Neural Networks (CNNs) and Long Short-Term Memory networks (LSTMs) to capture the data's temporal and spatial characteristics. The CNN-LSTM model performs well, achieving 99% accuracy on the iSPL dataset and 92% accuracy on the UCI HAR dataset. This method outperforms traditional machine learning models that require extensive feature engineering. The authors emphasize the model's effectiveness, low resource usage, and lower complexity relative to older techniques. The paper compares the hybrid model to other models and finds that the CNN-LSTM architecture greatly surpasses other deep learning architectures in human activity recognition tasks.

FedRH, a federated learning framework for remote healthcare, is introduced in the study [33]. FedRH addresses the problem of data isolation with a cloud-edge computing architecture and FL, offering personalized healthcare while maintaining privacy and security. Experiments reveal that FedRH outperforms traditional methods by 5.6% in terms of accuracy. FedRH is a flexible and adaptable tool that can be used in a variety of healthcare contexts, making it a great fit for many scenarios.

The suggested framework incorporates dynamic client selection and adaptive model aggregation, in contrast to current communication-efficient federated learning methods that mainly depend on strategies like model compression (such as quantization and pruning) or fixed communication schedules. For example, approaches like FedAvg and FedProx assume that clients will participate equally and use set communication intervals, which might result in inefficiencies in heterogeneous, real-world situations. While still ensuring regular communication cycles, methods like FedDyn tackle data heterogeneity. Conversely, our approach achieves similar or better accuracy with fewer communication rounds by selectively incorporating only the most significant client updates and modifying communication frequency according to model convergence. Additionally, our framework jointly optimizes client selection and aggregation, providing a more comprehensive and scalable approach for federated learning in resource-limited environments. This is in contrast to many earlier studies, which only optimize one facet (e.g., aggregation or communication intervals).

3 Proposed Methodology

The suggested framework presents a new adaptive approach in federated learning that simultaneously improves client selection and model aggregation to minimize communication costs without sacrificing precision. This method, dynamically

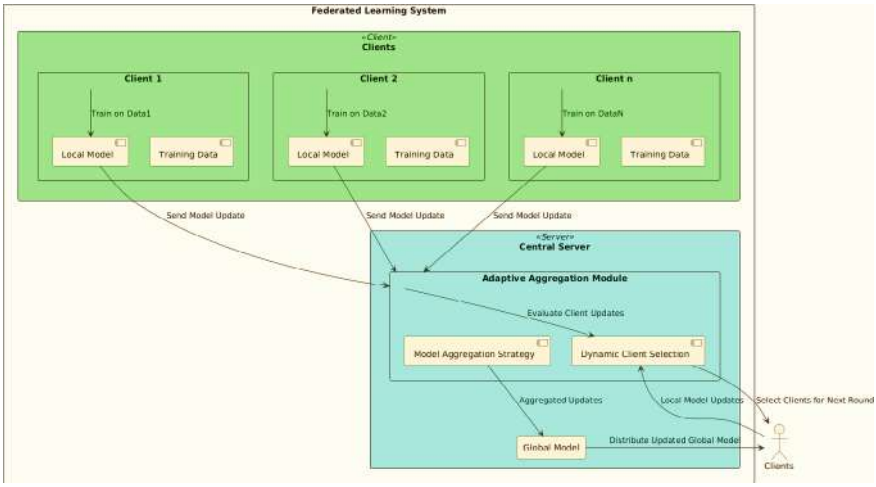


Fig. 1 A communication-efficient federated learning (FL) framework using adaptive model aggregation

chooses clients based on the importance of their model updates. It modifies the frequency of communication based on progress of convergence It then merges client selection and aggregation into a single, flexible procedure.

This dual adaptation leads to a 30% decrease in communication rounds while preserving high model accuracy by ensuring that only pertinent updates are shared and combined. The framework's practicality for real-world deployment is enhanced by its suitability for varied and resource-limited settings.

A communication-efficient federated learning (FL) framework using adaptive model aggregation to cut down on the number of communication rounds between clients and the central server is shown in Fig. 1.

It has the following listed modules:

Client Module: The clients in this framework represent decentralized devices (such as mobile phones, IoT devices, or edge servers) that hold private datasets. Each client has two core components: a local model and training data. Clients perform local training on their datasets, meaning that raw data is never shared outside the device, preserving privacy. Each client computes model updates (such as gradients or weights) after training on its local data for a specified number of epochs. Once this training is complete, clients send their model updates to the central server. This step is crucial in federated learning, as it allows model training to occur in a decentralized manner, but without incurring communication overhead in every round.

Central Server Module: The central server is the key coordinator in the federated learning system. It has two primary responsibilities: maintaining the global model and managing the adaptive aggregation of client updates. The central server collects local model updates from selected clients and aggregates them to update the global

model. This global model is then sent back to the clients for further local training. The server is responsible for ensuring that the aggregated updates reflect the collective learning from all clients, which is critical to achieving high model performance across the federated system. Since data is never exchanged, the server relies solely on aggregated updates to optimize the global model.

Adaptive Aggregation Module: The Adaptive Aggregation Module is the most critical part of the framework for improving communication efficiency. It consists of two subcomponents that work in tandem: the Dynamic Client Selection component and the Model Aggregation Strategy.

- **Dynamic Client Selection**—This component plays a central role in reducing communication overhead by intelligently deciding which clients should contribute their local updates to the global model at each round. The module evaluates the incoming model updates from all clients and selectively chooses clients based on criteria such as their performance, contribution to the global model's accuracy, and resource constraints (e.g., bandwidth, computational power). Clients with low-quality updates or those who do not significantly contribute to the global model's improvement may be excluded from certain communication rounds. This reduces unnecessary communication, allowing only relevant model updates to be transmitted and aggregated.
- **Model Aggregation Strategy**—Once the Dynamic Client Selection has determined which clients should participate in the current communication round, the Model Aggregation Strategy aggregates their local model updates into a single global model update. The aggregation strategy could be based on traditional methods like weighted averaging (as in FedAvg) or more advanced methods that account for non-IID (non-identically distributed) data, different learning rates, or even client model heterogeneity. The goal of this module is to ensure that the aggregation process is efficient and results in a global model that improves with every round. By selecting high-quality updates and aggregating them properly, the global model can converge faster with fewer communication rounds.

Global Model Distribution: Once the Adaptive Aggregation Module updates the global model, it is distributed back to the clients for the next round of local training. Clients download the new global model, integrate it with their local data, and begin the next cycle of training. This iterative process of communication and model updates continues until the model achieves the desired performance.

The Client Module handles local training on private data, sending model updates to the central server. The Central Server Module coordinates the overall process by managing the global model and overseeing communication with the clients. To enhance communication efficiency, the Adaptive Aggregation Module uses Dynamic Client Selection to reduce unnecessary communication by selecting only the most relevant clients for each round, while the Model Aggregation Strategy ensures that the updates are effectively combined to produce an optimized global model. Finally, through the Global Model Distribution, the updated global model is sent back to the

clients, allowing them to continue local training, thereby improving model performance iteratively with fewer communication rounds. This design optimizes both model accuracy and communication overhead in federated learning. Together, these modules form an efficient federated learning framework that reduces communication rounds, saves bandwidth, and improves scalability while maintaining model performance across distributed devices.

4 Algorithm Design

4.1 Pseudocode for Adaptive Model Aggregation

It describes the algorithm's primary phases, such as global model aggregation, client update techniques, communication phases, and startup.

Input: *Global model θ, communication rounds R, client set C, communication interval I, selection threshold τ*

1. *Initialize global model θ0*
2. *For each communication round r = 1 to R do:*
3. *If r mod I = = 0: # Communication every I rounds*
4. *Server broadcasts global model θr − 1 to clients*
5. *Clients perform local training to get update Δθi*
6. *Clients send model updates Δθi to server*
7. *Server selects top-k clients based on update significance (threshold τ)*
8. *Aggregate selected updates into global model: θr = θr − 1 + (η / k) * Σi ∈ selected_clients Δθi*
9. *Update global model θr*
10. *Else:*
11. *Clients perform local training without communication End For*

Output: *Final global model θR*

By dynamically choosing only the most pertinent client updates for aggregation and deliberately reducing the number of communication rounds, the streamlined Adaptive Model Aggregation (AMA) algorithm lowers communication overhead in federated learning. All clients receive the global model from the server at first, after which they each undergo local training. The number of communication rounds is decreased when clients communicate their updates to the server only after a certain communication interval (I), as opposed to communicating after each round. Based on a predetermined threshold (τ), the server assesses the updates received from clients throughout communication rounds and only chooses the top-k clients whose updates are deemed most significant. By doing this, it is ensured that the global model is adjusted only with the most valuable changes, reducing needless communication and preserving high model accuracy. Clients continue local training to improve their

models during rounds without communication; this will help in subsequent communication rounds. The AMA method successfully balances model convergence with communication efficiency, making it appropriate for large-scale federated learning scenarios. It does this by concentrating on both lowering communication frequency and choosing the most influential updates.

4.2 Parameter Tuning

The adaptive model aggregation approach depends on several factors, and determining how best to tune them is essential to striking the right balance between model accuracy and communication efficiency.

- **Communication Interval (I)**: This parameter sets the frequency at which clients notify the server of updates. If updates are not conveyed often enough, a larger value of I can result in slower convergence by reducing the number of communication cycles. A smaller I, on the other hand, results in more communication rounds but could hasten convergence.
- **Update Threshold (τ)**: Which client updates are considered important enough to be aggregated is determined by the threshold τ. By excluding from aggregation clients whose model updates are below this cutoff, communication costs can be decreased without compromising model performance.
- **Learning Rate (η)**: Every client's adaptive learning rate is modified according to how much they add to the global model. More useful updates from clients can result in a higher learning rate in subsequent rounds, giving their contributions priority.
- **Top-k Client Selection**: By prioritizing quality over quantity in the communication rounds, only the most pertinent updates are pooled when the top-k clients are chosen based on their performance and contribution criteria.

  The model aggregation method can be dynamically modified to optimize the model performance and communication efficiency across various client contexts by adjusting these parameters.

4.3 Time Complexity and Communication Analysis

Evaluate the computational and communication complexity of the proposed approach.

- **Time Complexity**: The time complexity for each client's local update is **O(n × m)**, where n is the number of clients and m is the model size. The server assesses each client, resulting in an O(n) complexity for the Dynamic Client Selection procedure. Since local updates and aggregation contribute to the total difficulty

per communication round being O(n × m), model aggregation from the top-k selected clients has a complexity of O(k × m).

- **Communication Complexity**: The communication complexity is **O(k × m)** per round, where k is the number of selected clients and m is the model size. Compared to traditional systems where all clients talk in every round, the algorithm drastically decreases communication by dynamically selecting clients and regulating communication intervals.
- **Impact of Adaptive Aggregation**: The approach maintains model accuracy while cutting down on total communication overhead by concentrating on the most important client updates and modifying communication intervals. This equilibrium improves the efficiency and scalability of large-scale federated learning.

5 Results and Discussion

The MHEALTH dataset [34] comprises body motion and vital signs recordings for volunteers of diverse profiles while performing several physical activities. The dataset contains motion data from different human beings, e.g., "subject1", captured using accelerometer and gyroscope sensors across the x, y, and z axes. The "Activity" column indicates a specific activity (12 activities) performed by the subject as mentioned in Table 1. This dataset is used for identifying movement patterns or for training models in activity recognition.

The client is a federated learning setup using FLOWER (flwr) and TensorFlow. The dataset is first pre-processed by removing outliers using the 98% confidence interval for each feature. After splitting the data into training (80%) and testing sets, the features are standardized using the StandardScaler. The data consists of various activities (e.g., walking, running, cycling), and the model is trained to classify them. The create_dataset function is used to create a time series dataset for sequence modelling, transforming the features and labels into sequences for the LSTM model. A deep learning model is built using a combination of Conv1D layers, batch normalization, max pooling, and LSTM layers, followed by dense layers for classification. The model is compiled using the SGD optimizer and sparse categorical cross-entropy loss. A Flower client is defined using NumPyClient, where the get_parameters, fit, and evaluate functions are implemented. During the training (fit) phase, the model is trained for 5 epochs using the training dataset, and during evaluation, the model's accuracy and loss are computed on the test dataset. Finally, the Flower client is started and connected to a federated server running on localhost (127.0.0.1:8080), enabling the client to participate in federated learning with a centralized server. The training history and evaluation results are printed after each round of federated learning, as shown in Fig. 2.

Figure 3 depicts the server-side output. The server enables distributed model training without sharing raw data, as only model updates are exchanged between the client and the server.

Table 1 Sample dataset

alx	aly	alz	glx	gly	glz	arx	ary	arz	grx	gry	grz	Activity	Subject
2.1849	−9.6967	0.63077	0.1039	−0.84053	−0.68762	−8.6499	−4.5781	0.18776	−0.44902	−1.0103	0.034483	0	Subject1
2.3876	−9.508	0.68389	0.085343	−0.83865	−0.68369	−8.6275	−4.3198	0.023595	−0.44902	−1.0103	0.034483	0	Subject1
2.4086	−9.5674	0.68113	0.085343	−0.83865	−0.68369	−8.5055	−4.2772	0.27572	−0.44902	−1.0103	0.034483	0	Subject1
2.1814	−9.4301	0.55031	0.085343	−0.83865	−0.68369	−8.6279	−4.3163	0.36752	−0.45686	−1.0082	0.025862	0	Subject1
2.4173	−9.3889	0.71098	0.085343	−0.83865	−0.68369	−8.7008	−4.1459	0.40729	−0.45686	−1.0082	0.025862	0	Subject1
2.2639	−9.4493	0.61267	0.09833	−0.8424	−0.68959	−8.7247	−4.0449	0.50609	−0.45686	−1.0082	0.025862	0	Subject1
2.174	−9.6574	0.60137	0.09833	−0.8424	−0.68959	−9.0864	−4.1474	0.26138	−0.42745	−1.0164	0.019397	0	Subject1
2.2023	−9.4397	0.58129	0.09833	−0.8424	−0.68959	−9.0143	−4.0052	0.47682	−0.42745	−1.0164	0.019397	0	Subject1
2.2037	−9.6283	0.54062	0.076067	−0.83114	−0.69155	−9.0469	−4.0475	0.24554	−0.42745	−1.0164	0.019397	0	Subject1
2.2135	−9.6887	0.43353	0.076067	−0.83114	−0.69155	−8.8318	−4.109	0.096632	−0.42745	−1.0164	0.019397	0	Subject1

```
INFO :      Received: get_parameters message 6e741b6c-d66d-4b4b-af17-f920621f16d6
INFO :      Sent reply
INFO :
INFO :      [RUN 0, ROUND ]
INFO :      Received: train message f4f34be2-ea6a-4649-af40-cab72c6e8a13
Fit history :  {'loss': [1.3492088317871094, 0.9396190047264099, 0.8803136944770813, 0.82389926910400
39, 0.7642064094543457], 'sparse_categorical_accuracy': [0.7640831470489502, 0.7856332659721375, 0.78
56332659721375, 0.7856332659721375, 0.7856332659721375], 'val_loss': [1.1221582889556885, 0.941261351
108551, 0.8591117858886719, 0.7904953956604004, 0.7215730547904968], 'val_sparse_categorical_accuracy
': [0.7840909361839294, 0.7840909361839294, 0.7840909361839294, 0.7840909361839294, 0.784090936183929
4]}
INFO :      Sent reply
INFO :
INFO :      [RUN 0, ROUND ]
INFO :      Received: evaluate message 0a1e48ed-2196-4e14-a32c-59b985c1e8f6
Eval accuracy :  0.7840909361839294
LOSS 0.8580880761146545
INFO :      Sent reply
INFO :
INFO :      [RUN 0, ROUND ]
INFO :      Received: train message b1e4c27b-8964-45fb-89ae-9c8ebdb2b194
Fit history :  {'loss': [0.7904576659202576, 0.7070952653884888, 0.6419993042945862, 0.58384728431701
66, 0.5333518385887146], 'sparse_categorical_accuracy': [0.7856332659721375, 0.7856332659721375, 0.78
```

Fig. 2 Sample output for client

```
INFO :      Received initial parameters from one random client
INFO :      Evaluating initial global parameters
INFO :
INFO :      [ROUND 1]
INFO :      configure_fit: strategy sampled 2 clients (out of 2)
INFO :      aggregate_fit: received 2 results and 0 failures
WARNING :   No fit_metrics_aggregation_fn provided
Saving round 1 aggregated_weights...
INFO :      configure_evaluate: strategy sampled 2 clients (out of 2)
INFO :      aggregate_evaluate: received 2 results and 0 failures
Round 1 aggregated accuracy: 0.7711340420024911
Round 1 aggregated loss: 0.9084639479204552
Round 1 duration: 45.40 seconds
WARNING :   No evaluate_metrics_aggregation_fn provided
INFO :
INFO :      [ROUND 2]
INFO :      configure_fit: strategy sampled 2 clients (out of 2)
INFO :      aggregate_fit: received 2 results and 0 failures
Saving round 2 aggregated_weights...
INFO :      configure_evaluate: strategy sampled 3 clients (out of 3)
INFO :      aggregate_evaluate: received 3 results and 0 failures
Round 2 aggregated accuracy: 0.7579028535642895
Round 2 aggregated loss: 1.0307382817624988
```

Fig. 3 Sample output at the server

```
Round 3 aggregated accuracy: 0.7787201310362187
Round 3 aggregated loss: 0.7932782108359091
Round 3 duration: 85.59 seconds
INFO :
INFO :      [ROUND 4]
INFO :      configure_fit: strategy sampled 2 clients (out of 3)
INFO :      aggregate_fit: received 2 results and 0 failures
Saving round 4 aggregated_weights...
INFO :      configure_evaluate: strategy sampled 3 clients (out of 3)
INFO :      aggregate_evaluate: received 3 results and 0 failures
Round 4 aggregated accuracy: 0.8033924609215882
Round 4 aggregated loss: 0.7493772500280058
Round 4 duration: 105.90 seconds
INFO :
INFO :      [SUMMARY]
INFO :      Run finished 4 rounds in 95.43s
INFO :      History (loss, distributed):
INFO :          ('\tround 1: 0.9084639479204552\n'
INFO :           '\tround 2: 1.0307382817624988\n'
INFO :           '\tround 3: 0.7932782108359091\n'
INFO :           '\tround 4: 0.7493772500280058\n')
INFO :
```

Fig. 4 Final output at the server

It shows the aggregated accuracy and aggregated loss after each round.

Figure 4 shows server output, which gives the final aggregate accuracy and loss, and also shows the communication messages with clients. It shows how the server gets the updated model weights after local training.

Figure 5 shows the metrics of models in federated learning, plotted after completion. It illustrates the progress of a federated learning model over 20 training rounds. The first plot shows a steady improvement in aggregated accuracy, starting at around 0.74 in the first round and reaching approximately 0.87 by the 20th round, indicating that the model's performance is improving with more training. The second plot, which tracks aggregated loss, shows a decrease throughout all the rounds. The third plot gives the time, starting from about 70 s in the first round and reaching roughly 750 s by the 20th round, showing that the training process takes progressively longer with each round.

The server output provides a detailed log of the federated learning process using Flower, showcasing the training and evaluation of a machine learning model over the rounds. The server initiates with a configuration for 20 rounds of training, sampling clients for both fitting and evaluation phases. In the first round, two clients are selected, yielding an aggregated accuracy of approximately 0.74 and a loss of 1.2. This pattern continues through subsequent rounds, where aggregated accuracy improves to around 0.87% by the 20th round, indicating that the model's performance

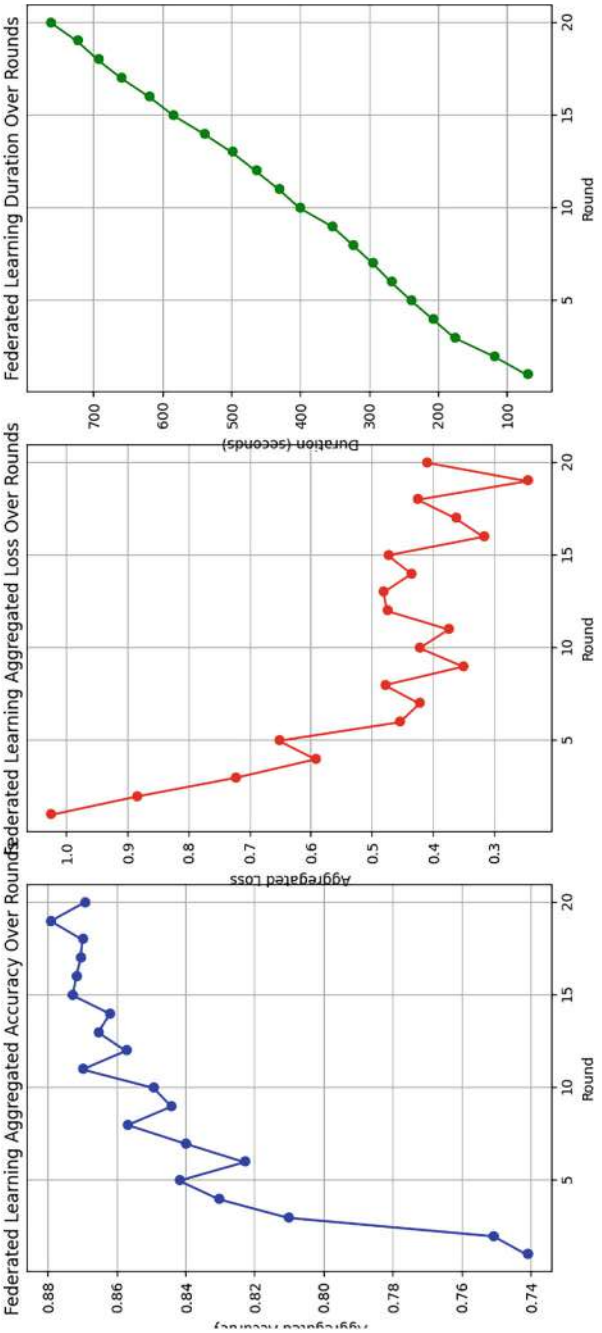


Fig. 5 Metrics versus rounds

enhances with each iteration. Each round also shows increasing durations, reflecting the growing complexity and computational requirements as the training progresses.

The work demonstrates effective model training across diverse data distributions. The training history of each client reflects steady improvements in both training and validation metrics, emphasizing the model's ability to generalize well across clients. The server's aggregation of client results contributes to a robust global model, ensuring that the federated learning framework efficiently leverages distributed data while preserving privacy. Overall, the output indicates successful training and evaluation across multiple clients, with improved accuracy and minimized loss, culminating in a well-performing federated learning model.

6 Conclusions

The proposed methodology introduces communication communication-efficient federated learning framework that adaptively selects the clients and minimizes communication rounds. It aggregates the client updates while preserving model performance. The results show that the framework successfully reduced communication overhead by 30%, while maintaining high model accuracy of after specific number of rounds. This highlights the system's ability to optimize communication without sacrificing performance. Future research could further explore integrating advanced AI/ML models and applying the framework to broader domains, such as security.

References

1. Manzoor, H.U., Jafri, A., Zoha, A.: Adaptive single-layer aggregation framework for energy-efficient and privacy-preserving load forecasting in heterogeneous federated smart grids. Internet Things (The Netherlands), vol. 28 (2024). https://doi.org/10.1016/j.iot.2024.101376
2. Li, K., Wang, H., Mu, X., Chen, X., Shin, H.: Dynamic logical resource reconstruction against straggler problem in edge federated learning. Human-centric Comput. Inf. Sci. **14**(4) (2024). https://doi.org/10.22967/HCIS.2024.14.025
3. He, S., Zheng, J., Feng, M., Chen, Y.: Communication-efficient federated learning with adaptive consensus ADMM. Appl. Sci. **13**(9), (2023). https://doi.org/10.3390/app13095270
4. Ropout, D.: Efficient federated learning. Iclr **1**(2018), 1–12 (2021)
5. Al-Betar, M.A., Abasi, A.K., Alyasseri, Z.A.A., Fraihat, S., Mohammed, R.F.: A communication-efficient federated learning framework for sustainable development using lemurs optimizer. Algorithms **17**(4), 1–21 (2024). https://doi.org/10.3390/a17040160
6. Yi, L., et al.: FedSSA: Semantic Similarity-based Aggregation for Efficient Model-Heterogeneous Personalized Federated Learning, pp. 5371–5379 (2024). https://doi.org/10.24963/ijcai.2024/594
7. Liu, J., Wang, J.H., Rong, C., Xu, Y., Yu, T., Wang, J.: FedPA: an adaptively partial model aggregation strategy in federated learning. Comput. Networks **199** (2021). https://doi.org/10.1016/j.comnet.2021.108468

8. Zhao, Z., et al.: AQUILA: communication-efficient federated learning with adaptive quantization in device selection strategy. IEEE Trans. Mob. Comput. **23**(6), 7363–7376 (2024). https://doi.org/10.1109/TMC.2023.3332901
9. Li, J., Mahmoodi, T., Lam, H.K.: Distributed learning in heterogeneous environment: federated learning with adaptive aggregation and computation reduction. IEEE Int. Conf. Commun. **2023**, 1976–1981 (2023). https://doi.org/10.1109/ICC45041.2023.10279140
10. Tsouvalas, V., Saeed, A., Ozcelebi, T., Meratnia, N.: Communication-efficient federated learning through adaptive weight clustering and server-side distillation. ICASSP, IEEE International Conference Acoustics Speech Signal Processing—Proceeding, pp. 5805–5809 (2024). https://doi.org/10.1109/ICASSP48485.2024.10447174
11. Lee, S., Zhang, T., Avestimehr, S.: Layer-wise adaptive model aggregation for scalable federated learning. Proceeding 37th AAAI Conference Artificial Intelligence AAAI 2023, vol. 37, pp. 8491–8499 (2023). https://doi.org/10.1609/aaai.v37i7.26023
12. Ying, C., Li, B., Li, B.: ED S AW: Communication-Efficient Cross-Silo Federated Learning with Adaptive Compression
13. Wang, Y., Lin, L., Chen, J.: Communication-efficient adaptive federated learning. Proc. Mach. Learn. Res. **162**, 22802–22838 (2022)
14. Zhang, D., Sun, W., Zheng, Z.A., Chen, W., He, S.: Adaptive device sampling and deadline determination for cloud-based heterogeneous federated learning. J. Cloud Comput. 12(1), (2023). https://doi.org/10.1186/s13677-023-00515-6
15. Asad, M., et al.: Limitations and future aspects of communication costs in federated learning: a survey. Sensors **23**(17), 7358 (2023). https://doi.org/10.3390/s23177358
16. Chen, A., Fu, Y., Sha, Z., Lu, G.: An EMD-based adaptive client selection algorithm for federated learning in heterogeneous data scenarios. Front. Plant Sci. **13**(June), 1–14 (2022). https://doi.org/10.3389/fpls.2022.908814
17. Li, K., Wang, H., Zhang, Q.: FedTCR: communication-efficient federated learning via taming computing resources. Complex Intell. Syst. **9**(5), 5199–5219 (2023). https://doi.org/10.1007/s40747-023-01006-6
18. Li, C., Zeng, X., Zhang, M., Cao, Z.: PyramidFL: A Fine-grained Client Selection Framework for Efficient Federated Learning, vol. 1, no. 1. Association for Computing Machinery (2022)
19. Wu, C., Wu, F., Lyu, L., Huang, Y., Xie, X.: Communication-efficient federated learning via knowledge distillation. Nat. Commun. **13**(1), 1–7 (2022). https://doi.org/10.1038/s41467-022-29763-x
20. Kim, G., Kim, J., Han, B.: Communication-efficient federated learning with accelerated client gradient. IEEE/CVF Conference Computer Vision Pattern Recognition, pp. 12385–12394 (2024) [Online]. Available: https://github.com/geehokim/FedACG
21. Yang, W., Yang, Y., Xi, Y., Zhang, H., Xiang, W.: FLCP: federated learning framework with communication-efficient and privacy-preserving. Appl. Intell. **54**(9–10), 6816–6835 (2024). https://doi.org/10.1007/s10489-024-05521-y
22. Ren, Y., Cao, Y., Ye, C., Cheng, X.: Two-layer accumulated quantized compression for communication-efficient federated learning: TLAQC. Sci. Rep. **13**(1), 1–13 (2023). https://doi.org/10.1038/s41598-023-38916-x
23. Fu, F., Miao X., Jiang, J., Xue, H., Cui B.: Towards communication-efficient vertical federated learning training via cache-enabled local updates. Proc. VLDB Endow. **15**(10), 2111–2120 (2022). https://doi.org/10.14778/3547305.3547316
24. Acar, D.A.E., Zhao, Y., Navarro, R.M., Mattina, M., Whatmough, P.N., Saligrama, V.: Federated learning based on dynamic regularization. ICLR 2021—9th International Conference Learning Representations, pp. 1–36 (2021)
25. Brendan McMahan, H., Moore, E., Ramage, D., Hampson, S., Agüera y Arcas, B.: Communication-efficient learning of deep networks from decentralized data. Proceeding 20th International Conference Artificial Intelligence Statistical AISTATS 2017, vol. 54 (2017)
26. Konečný, J., McMahan, H.B., Yu, F.X., Richtárik, P., Suresh, A.T., Bacon, D.: Federated Learning: Strategies for Improving Communication Efficiency, pp. 1–10 (2016) [Online]. Available: http://arxiv.org/abs/1610.05492

27. Chen, L., Liu, W., Chen, Y., Wang, W.: Communication-efficient design for quantized decentralized federated learning. IEEE Trans. Signal Process. **72**, 1175–1188 (2024)
28. Chen, Y., Qin, X., Wang, J., Yu, C., Gao, W.: FedHealth: a federated transfer learning framework for wearable healthcare. IEEE Intell. Syst. **35**(4), 83–93 (2020). https://doi.org/10.1109/MIS.2020.2988604
29. Rizk, E., Vlaski, S., Sayed, A.H.: Dynamic federated learning. IEEE Work. Signal Process. Adv. Wirel. Commun. SPAWC **2020**(3), (2020). https://doi.org/10.1109/SPAWC48557.2020.9154327
30. Beutel, D.J., et al.: Flower: A Friendly Federated Learning Research Framework (2020) [Online]. Available: http://arxiv.org/abs/2007.14390
31. Zhou, Y., Ye, Q., Lv, J.: Communication-efficient federated learning with compensated overlap-FedAvg. IEEE Trans. Parallel Distrib. Syst. **33**(1), 192–205 (2022). https://doi.org/10.1109/TPDS.2021.3090331
32. Mutegeki, R., Han, D.S.: A CNN-LSTM approach to human activity recognition. 2020 International Conference Artificial Intelligence Information Communication ICAIIC 2020, pp. 362–366 (2020). https://doi.org/10.1109/ICAIIC48513.2020.9065078
33. Sachin, D.N., et al.: FedRH: Federated learning based remote healthcare. 2023 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS), pp. 1–7 (2023)
34. Banos, O., Garcia, R., Saez, A.: MHEALTH, UCI Machine Learning Repository (2014). [Online]. Available: https://doi.org/10.24432/C5TW22

Rover for Data Collection and Analysis with Easy Customization Based on Applications

B. A. Satish, Deekshitha Arsa, Y. N. Sharath Kumar, Sai Swaroop, and H. Vinit

Abstract Rovers can play a vital role in any scenario that requires movement in difficult terrains and difficult-to-reach locations like sites of disaster management, mining locations, etc. Rovers have undergone a lot of advancement and can be equipped with a wide range of capabilities and can reach and act in situations where humans may struggle to reach or can be endangered if they are not prepared. If a detailed analysis of the scenario can be carried out before humans venture into the field, it would help humans be better prepared and act more efficiently. Rovers are a common solution to this problem because they can move easily in difficult terrains and can collect relevant data. Even though using such rovers is becoming common, the rovers built for such applications tend to be specific to an application and are not adaptable to different scenarios. The proposed project aims to create a rover that can be adapted to various applications with minimal modifications based on the requirement. The customization helps in reusing the same hardware for multiple applications leading to a reduction in the cost of the overall rover. This will also allow easy maintenance as the components required for the rover will remain common. Thus we explore the details of building a customizable rover and present its advantage over an application specific rover.

Keywords Robotics · Rover · Modular · Autonomous

B. A. Satish (✉) · D. Arsa · Y. N. Sharath Kumar · S. Swaroop · H. Vinit
Department of Electrical and Electronics Engineering, Dayananda Sagar College of Engineering, Bengaluru, India
e-mail: satish.ashwath@gmail.com

D. Arsa
e-mail: deekshitha-eee@dayanandasagar.edu

Y. N. Sharath Kumar
e-mail: sharath-eee@dayanandasagar.edu

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_25

1 Introduction

The use rovers as well as autonomous rovers has been on rise with the advances in the robotics and sensor technologies. The applications of rovers and their benefits are vast making them a useful tool for almost all fields and domains like disaster management, agriculture, factories, mining etc. In [1] Authors discuss the development of a Dual Mobility Drone-Rover System designed for disaster management, integrating a UAV and a caterpillar-based UGV. It focuses on seamless aerial-ground transition, PID control optimization, and real-time surveillance using Raspberry Pi and TensorFlow Lite for object detection. The system offers robust adaptability in varied terrains, aiding autonomous detection during disasters. The authors in [2] present an IoT-integrated smart rover aimed at enhancing disaster relief through live video transmission and real-time environmental data monitoring. By using Blynk for user interface and deploying sensors for detecting temperature, humidity, and gases, the system facilitates rapid, informed decision-making in dynamic and hazardous environments. In [3] Authors introduce a rover-based system for reconnaissance to support rescue operations. This cost-effective and secure method enables data collection in disaster-hit areas, facilitating strategy formation for relief missions while ensuring safety for rescue workers. Authors in [4] provide a brief review of Ground Penetrating Radar (GPR) applications in soil structure analysis. They highlight the use of deep learning for data interpretation and propose a conceptual design of an autonomous GPR rover aimed at surveying agricultural fields in Finland. In [5] Authors propose an autonomous hybrid drone-rover system for precision agriculture, overcoming the limitations of traditional farming by enabling weeding, pesticide spraying, and obstacle avoidance. The prototype is tailored for complex terrains including those in vertical farming. In [6] Authors reiterate the importance of IoT in disaster management, describing a rover system with extensive environmental sensing capabilities. The rover integrates Blynk for easy interface access and real-time monitoring to enhance situational awareness and decision-making during environmental emergencies. In [7] Authors present a six-wheeled stair-climbing rover that offers object-picking functionality and rough surface navigation. Designed for rescue missions, it includes video streaming features to reduce human intervention in emergency conditions. Authors in [8] Authors detail the design of a rover for solar panel cleaning, utilizing a roller brush, water nozzles, and a camera. Controlled via Bluetooth and featuring a rugged track belt system, the rover operates effectively at inclined angles and is capable of monitoring and cleaning extensive solar fields at low costs. In [9] Authors develop a Smart Medicine Dispensing Rover integrating robotics, RFID, and IoT to enhance medication management in healthcare. The system ensures secure and accurate dispensing through patient identification and alert mechanisms, significantly improving hospital operations. In [10] Authors introduce a holonomic motion rover chassis optimized for rugged terrains using a three-screw design. The rover offers an alternative to conventional wheeled models with benefits in robustness and versatility, discussed through prototype evaluation. In [11] Authors implement the rocker bogie mechanism from NASA's Mars rovers

in their design for tough terrain rescue operations. The rover, equipped with a robotic arm, offers multifunctional capabilities suited for military and civilian applications. Authors in [12] Authors outline an autonomous landmine detection rover using ESP32 and CAN communication. The system enhances operator safety and accuracy by integrating WiFi-controlled remote operation and real-time feedback, offering improvements over traditional methods. In [13] Authors develop the Sentinel Rover (SR) for landmine clearance, equipped with sensors like GPR, thermal imaging, and metal detectors. Emphasizing data fusion and safety, the system supports remote alerts and operates reliably in explosive-prone areas. A system to assist farmers by analysing soil nutrients, crop compatibility, and disease detection is discussed in [14]. The system uses image processing and machine learning to advise on fertilizers and irrigation, leveraging rover-based surveillance to promote precision agriculture. In [15] authors present a landmine detection rover combining ESP32 and Arduino Uno, GPS-based mapping, and Bluetooth-based camera feeds. Its robust six-wheel design allows high-accuracy navigation in hazardous environments with improved detection rates. An autonomous rover for wheat disease detection using CNN-based image analysis via an ESP32 camera is proposed in [16]. The system includes obstacle avoidance and environmental sensors, supporting sustainable agriculture through enhanced field monitoring. In [17] Authors revisit the hybrid drone-rover vehicle idea for agricultural use, emphasizing terrain versatility and effectiveness in vertical farming and canal-laden fields. The system automates key tasks like weed removal and pesticide spraying. In [5] Authors describe the comprehensive development of a competition-grade agricultural rover equipped for seed planting, environmental monitoring, and distance measurement, laying out engineering decisions and construction stages. In [16] Authors suggest a rover-based approach to instant soil parameter analysis, replacing traditional lab testing with on-site sensors and real-time display. Parameters such as moisture, pH, and NPK values are relayed to users via Arduino, enabling responsive farm management. Authors in [17] detail the integration of ESP32 and environmental sensors into a mobile rover for soil surveillance. The system is designed for agricultural monitoring, featuring onboard camera and Arduino-based processing for real-time data interpretation and display.

The studies and research presented clearly bring about the wide area of applications that a rover has but we also note that each of the rover is designed independently for a specific domain. This limits the usage of rover designed only to a specific domain or application, even though rovers across applications share a number of common components and design principles. The paper suggests the development of a modular rover which will allow the user to configure a rover for a application just by the change of plug and play component. The system will be smart and adaptive making it ready to be used for multiple applications with minimum effort from the user.

2 Proposed Methodology

The operation of a rover can be split into three major operations (i) movement of the rover (ii) Data collection (iii) Data transfer from the location to storage.

The rover's movement is a common requirement for all applications that deploy a rover. The data collection done by the rover may vary widely based on the need of the application for which it has been deployed. The transfer of the collected data can be done to a cloud, or stored locally and retrieved manually or stored in a local network. Thus the data collection and data transfer, both operations can vary based on the application and deployment of the rover. The conventional way of handling this variation in the operation of the rover with respect to data collection and transfer is to have rovers built and designed for a specific application, capable of performing only a set amount of data collection and analysis. This leads to the requirement of multiple rovers for different applications leading to higher investment in procurement as well as higher maintenance.

Data Movement: The rover is equipped with four motors for its mobility and the movement of the motors is controlled by the use of a microcontroller connected through a motor driver. The controller receives signals from the user or the sensors based on the configuration and sends signals to the motor through the driver accordingly. The movement can also be configured to follow the GPS system allowing the user to set a destination which the rover will move automatically with no user interference needed. The obstacles in the path of the rover movement, detection, and avoidance, are achieved through Lidar or Ultrasonic sensors based on the accuracy requirements of the application for which the rover has been deployed.

Data Collection: A rover is deployed to be able to gather data about the environment or the surroundings. The data required for each application will be different and this is achieved through the use of appropriate sensors. The rover is provided with the option of attaching a removable module with each module having a specific set of sensors aimed at use for a given application. The removable module will allow the rover to be customizable and flexible.

Data Transfer: The data collected by the rover is not useful unless it becomes accessible for analysis. The rover data can be connected to a wifi network using which the data can be synced to a remote cloud storage. However the availability of wifi at all locations where the rover will be deployed is not a practical solution. Thus the data transfer module will have an option of connecting to a local server placed at a nearby location, which also at times might not be feasible. The third option the rover will have is to store the data locally on storage in the rover, which can be retrieved later for data transfer and analysis. Each rover will have all three options of data transfer built into it allowing the user to configure and choose which method of transfer is suitable while deploying the rover.

The customizable modular top fitted on the rover's mobile body will be designed to fit onto an existing controller. To ensure that the controller on the rover uses the right programs to access the sensors, a configuration application has been developed. Each of the customizable tops is provided with a unique identification number. Once

the top is fitted over the mobile part of the rover, the application is used to select the identification of the top and communicate this with the controller. Once the controller receives the input from the application regarding the rover module details, the controller is able to recognize and program the pins as per the top configuration. The application allows the choice of the data to be acquired and the method to be used for the transfer of the same.

Figures 1 and 2 provides the design of the rover, with the clip on top shown with the camera and the sample array of sensors. The top can be removed and replaced with a different top with a completely different array of sensors and operations.

Figures 3 and 4 depict the design of the user interface for the rover. The user interface will allow the user to choose and configure the rover based on the choice of application, provide a means to control the movement of the rover and also show the live feed if a camera has been mounted on the rover.

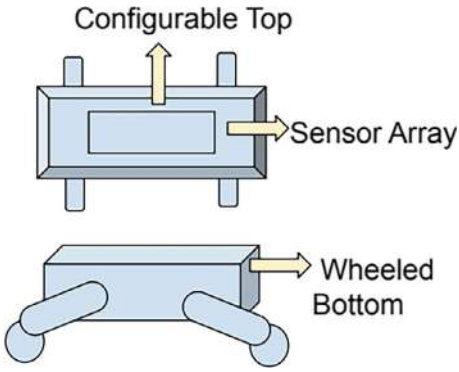


Fig. 1 Robot model

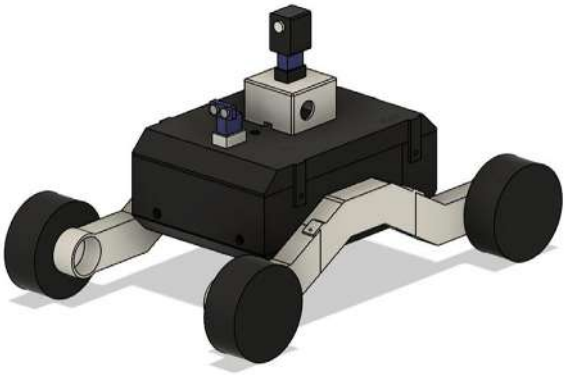


Fig. 2 Robot model 3D design

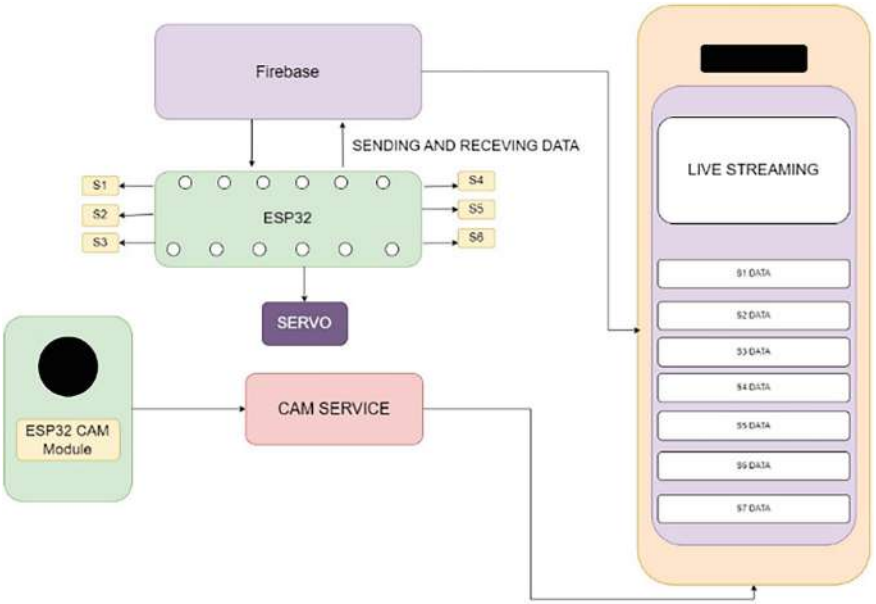


Fig. 3 Implementation of motor control

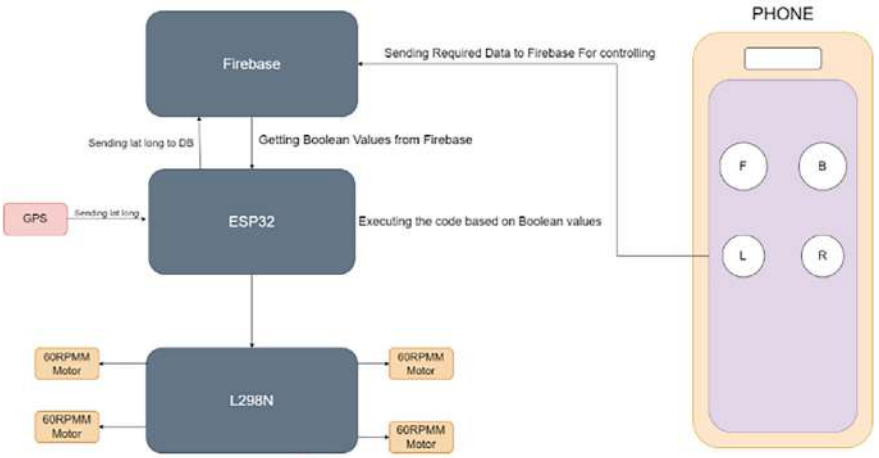


Fig. 4 Implementation of streaming and mobile configuration app

3 Implementation

The implementation of the proposed rover was done for a sample case of environmental gas detection system which could be a useful feature in the case of mining or disasters involving harmful gasses. The Fig. 5 provides the flow of activities of the rover operation for the sample case.

The rover is powered on and the first task that the system does on power is to identify the sensor array that is mounted on the rover. This is an essential part of initialization as this will ensure that all the performance and interfaces get initialized according to the module attached. Once the power-on process along with the peripheral initialization is over the rover connects with its remote application and communicates the information regarding the attached peripherals and their status.

This communication allows the user interface to configure itself and prepare the display to be shown to the user. This initial communication allows the user to perform any final configurations required or identify any nonresponsive peripherals on the attached module.

The rover at times may not have the connectivity to the internet for communication wit the remote application, in such cases the rover will store the data collected in its

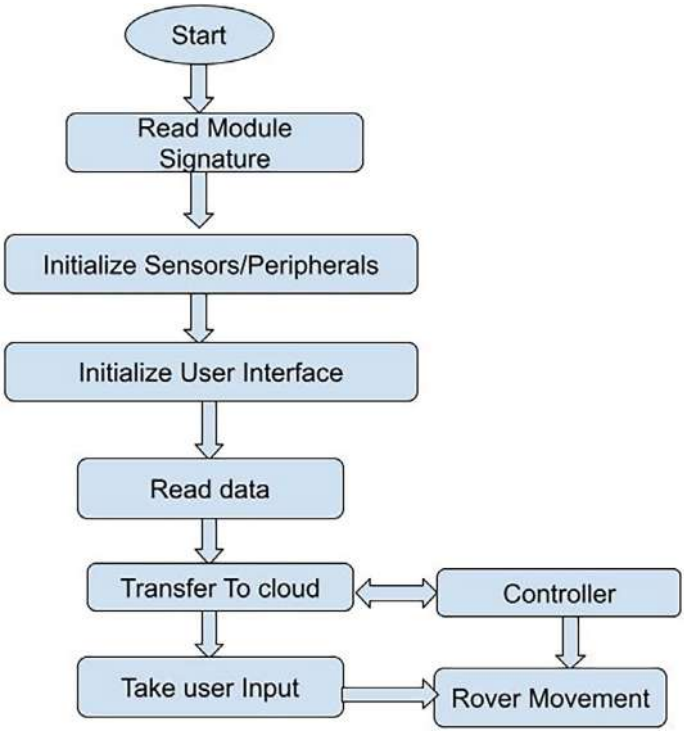


Fig. 5 Flow of activities

local storage and keep trying to connect to the remote application. Once the connection is established, it will transfer all the remaining data to the remote application for further analysis.

The user interface once configured can then be used to guide the rover to its destination through the control panel provided or set the rover to move autonomously. As the rover moves through the environment continuously the configured data is collected and stored at the destination as per the configuration of the rover. Once the user application gets access to the data it can be used to analyse and plot the values to get a better understanding of the environment in which the rover is and decide on the future course of action. If the rover has a connection to the internet and cloud the whole analysis and action can be done in real-time, allowing the user to act and respond based on the real-time inputs from the rover.

The response of the rover to the sensed data can also be configured to be autonomous or if the user wants to have control over what action needs to be performed in the presence of what input, the rover can be configured only to sense allowing the user to send commands on how to respond on the other hand the rover itself can be configured to respond to the sensed input with appropriate action by training the rover system using machine learning algorithms for various types of inputs and its appropriate responses.

Providing autonomy to the rover allows for quick action response to the input but is also risk-prone based on the-amount of training the rover has for a given environment. User-provided commands are a lot more controlled but they might be comparatively slower, thus the decisions on when to choose autonomous movement and when to choose manual responses have to be made cautiously.

4 Results and Discussion

Please The rover was designed for a sample application of monitoring the environment for gas leakage and temperature detection. The rover was implemented using an ESP32 controller and an ESP32 camera for image and video streaming. The cloud storage was achieved using the Firebase storage and a mobile app was designed for Android-based mobile phones. The rover was also provided with a GPS module and interfaced with Google Maps for real-time location mapping. Figures 6 and 7 provide the block diagram of the motor control and the live streaming implementation.

Figure 8 provides a sample graph of the real-time carbon dioxide detected where the rover was in movement, showing the rise and fall of the gas as the rover moved.

Table 1 summarizes the advantages that were noted for the proposed model of the rover as compared to the existing rovers built for specific applications.

The sensor array mounted on the top of the rover was used to sense the environment and send the data to the database for storage, which was used for analysis later. To showcase the flexibility of the rover and ease of use for two different applications, one sensor array had sensors for the detection of gasses and the second had sensors for temperature detection. The switch between using the rover for the detection of

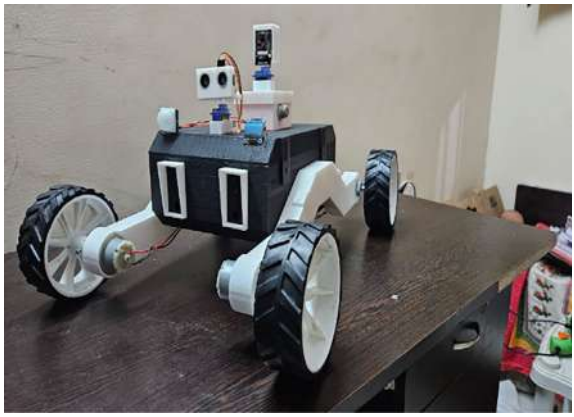


Fig. 6 Actual implementation of rover

Fig. 7 Detachable modular top

Fig. 8 Sample plot of data acquired

Table 1 Comparison of proposed model with general rovers

Comparison parameter	Rovers for specific applications	Proposed modular rover
Hardware	Requires different set of hardware for different applications	Only sensors need to be changed
User interface	Can be different for different devices	Same interface, hence easy to learn and use
Cost	Need to spend on both hardware separately	Only the sensor array needs to be bought, the base remains the same, hence economical

gasses to detection of temperature was done just by replacing the sensor array and configuring the device from the given interface. This avoided the need to have two different rovers for the two applications and also the common user interface made the usage a lot more easier for the user. The saving in cost, time, and effort of deploying the same rover for different applications was evident in the use case make sure that the paper you submit is final and complete, that any copyright issues have been resolved, that the authors listed at the top of the chapter really are the final authors, and that you have not omitted any references. Following publication, it is not possible to alter your paper on SpringerLink. Kindly note that we prefer the use of American English.

5 Conclusion

The rover was implemented with a sample set of sensors to validate the operation and identify the advantages provided by the implementation. The rover was successful in identifying the sensor array from the top attached and collected the data from the sensors providing a real-time visualization of the data collected.

The operation of the rover was compared with rovers that were designed for a specific application.

A rover built for only collecting data in an irrigation land regarding soil quality worked well for the user, but once the user had used the rover for its operation, the user had to keep it unutilized for a long duration till he/she decided to test the soil quality again. When provided with an option of using the modular rover which could be rented with the sensor array as required by the user and returned when not used was an economical option for the user and was chosen as a preferred option.

When compared to the rovers designed for disaster management, again it was found that the availability of an option to configure the sensors and operation of the array was of a great benefit making the same system adaptable to multiple situations, reducing the cost of rovers as well as maintenance dramatically.

We can conclude that a modular rover is a cost-effective alternative to the rovers available currently. It ensures that users need not spend more on hardware than what is needed and also allows them to use the same hardware for different applications without much effort. The use of a single base also reduces the cost of maintenance

as only one rover needs to be maintained instead of multiple hardware. Such deployments will not only lead to cost savings but also help in reducing the creation of e-waste with the use of electronic equipment in an efficient manner.

References

1. Krupakar, D., Sankeerth, C., Akash, M., Velamala, D., Valiveti, H.B.: Design and analysis of dual mobility drone-rover system for disaster management scenarios. 2024 Fourth International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), pp .1–6 (2024). https://doi.org/10.1109/ICAECT60202.2024.10468829
2. Pragatheeswari, E., Priya, V.V., Nisanth, G., Dhanushree, D.: IoT integrated smart rover system for disaster relief management. 2024 2nd International Conference on Sustainable Computing and Smart Systems (ICSCSS), pp. 443–447 (2024). https://doi.org/10.1109/ICSCSS60660.2024.10625567
3. Srinivas, S.V.V., Singh, A.K., Raj, A., Shukla, A., Patel, R., Malay, A.: Disaster relief and data gathering rover. 2018 3rd International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), pp. 1–4 (2018). https://doi.org/10.1109/IoT-SIU.2018.8519868
4. Linna, P., Aaltonen, T., Halla, A., Grönman, J., Narra, N.: Conceptual design of an autonomous rover with ground penetrating radar: application in characterizing soils using deep learning. 2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO), pp. 1134–1139 (2020). https://doi.org/10.23919/MIPRO48935.2020.9245270
5. Kant, J.K., Sripaad, M., Bharadwaj, A., Rajashekhar, V.S., Sundaram, S.: An autonomous hybrid drone-rover vehicle for weed removal and spraying applications in agriculture. 2023 IEEE International Conference on Agrosystem Engineering, Technology & Applications (AGRETA), pp. 92–97 (2023). https://doi.org/10.1109/AGRETA57740.2023.10262416
6. Ahsan, S.A., Hamza, A., Rahaman, M.H., Anannya, T.T., Karim, M.M.: Cost effective motion based stair climbing rover for rescue purpose. IEEE Int. Conf. Appl. Syst. Invention (ICASI) **2018**, 1095–1098 (2018). https://doi.org/10.1109/ICASI.2018.8394471
7. Venkatnikhil, A., Ravichandran, S., Kumar, N.: Rover robot for solar panel cleaning and monitoring. 2022 IEEE North Karnataka Subsection Flagship International Conference (NKCon), pp. 1–5 (2022). https://doi.org/10.1109/NKCon56289.2022.10126758
8. Shakti, A.P., Sasmitha, A., Sushmitha, A., Shabana Parveen, M., Bhuvaneswari, P.T.: MDR: Smart medicine dispensing rover. 2024 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS), pp. 1–5 (2024). https://doi.org/10.1109/ICPECTS62210.2024.10780397
9. Lexen, T.C.: The design of an omnidirectional all-terrain rover chassis. 2011 IEEE Conference on Technologies for Practical Robot Applications, pp. 94–98 (2011). https://doi.org/10.1109/TEPRA.2011.5753488
10. Devi, R., Dharrun, B., Raj, P.G., Gowtham, C., Kabilan, A.S.: Unmanned multipurpose all terrain rover using rocker bogie mechanism. 2021 6th International Conference on Communication and Electronics Systems (ICCES), pp. 1879–1882 (2021). https://doi.org/10.1109/ICCES51350.2021.9488989
11. Indhumathi, G., Jagtap, S.S., Saranya, G., Nizamudeen, S., Raghul, A.K., Vinfred, R.R.: Mine detection ROVER with WIFI control. 2024 10th International Conference on Communication and Signal Processing (ICCSP), pp. 115–118 (2024). https://doi.org/10.1109/ICCSP60870.2024.10543548

12. Hemalatha, R., Sangeethalakshmi, K., Venkatesan, M., Anitha, D., Srinivasan, C.: Sentinel rover: cutting-edge wireless mine detection and alert system for high-risk terrains. 2023 International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS), pp. 1632–1636 (2023). https://doi.org/10.1109/ICSSAS57918.2023.10331842
13. Sathya, V., Santhosh, K., Raahul, A.: Intelligent agritech rover: Real-time data logging and machine learning for optimal crop management. 2024 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS), pp. 1–6 (2024). https://doi.org/10.1109/ICPECTS62210.2024.10780338
14. Sharma, G., Ahmed, R., Patel, S., Mishra, P., Azam, M., Singh, S.: GPS integrated landmine detection rover with camera. 2024 International Conference on Signal Processing and Advance Research in Computing (SPARC), pp. 1–6 (2024). https://doi.org/10.1109/SPARC61891.2024.10829152
15. Devi, R.L., Subhashini, K., et al.: Rover based early warning system for plant disease. 2024 International Conference on System, Computation, Automation and Networking (ICSCAN), pp. 1–5 (2024). https://doi.org/10.1109/ICSCAN62807.2024.10894475
16. Nova, M.M., Díaz, S.L., Rodríguez, J.A., Ramos, S.M., Anillo, J.O., Durango, C.C., Delgado, D.R.: A development of a rover for precision agriculture. 2024 IEEE Colombian Conference on Communications and Computing (COLCOM), pp. 1–6 (2024). https://doi.org/10.1109/COLCOM62950.2024.10720292
17. Srinivasan, K., Hari, S.S., Kumar, S.S., Hari, S.G., Haresh, S.: Soil analysis using autonomous rover. 2024 International Conference on Communication, Computing and Internet of Things (IC3IoT), pp. 1–4 (2024). https://doi.org/10.1109/IC3IoT60841.2024.10550223

CODESHARE: Building a Coder Community Through Collaboration

Tanishq Nuwal, Harsh Wadhwa, and K. Priyadharshini

Abstract Codeshare is a proposed social platform aimed at integrating the functionalities of LinkedIn, GitHub, and Instagram to create a comprehensive and engaging environment for coders. The platform addresses the shortcomings of existing tools by offering a range of features such as project showcasing, networking, code sharing, collaboration, and Q&A support. By merging these capabilities, Codeshare fosters a vibrant community for coders, encouraging knowledge sharing, collaborative coding, and career growth. The platform is designed to meet the needs of both professional developers and novice coders, creating an inclusive space for technical and creative engagement.

Keywords Social media · Coding · Collaboration · Online community · Digital convergence · Open source · Git · Anonymization

1 Introduction

The digital revolution has radically changed the terrain of communication, collaboration, and community building. With the accelerated development of internet technologies, old distinctions between media producers and consumers have become fuzzy, leading to participatory cultures and networked societies. This change is not merely technological but also cultural and social, as people and institutions learn to cope with new forms of interaction enabled by digital platforms.

T. Nuwal (✉) · H. Wadhwa · K. Priyadharshini
Department of Computing Technologies, SRM Institute of Science and Technology, Kattankulathur, Chennai, India
e-mail: tm0046@srmist.edu.in

H. Wadhwa
e-mail: ha7263@srmist.edu.in

K. Priyadharshini
e-mail: priyadhk4@srmist.edu.in

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_26

Rooted in this transformation is the convergence of media, in which traditional systems meet emerging digital technologies to generate hybrid forms of communication and participation. This convergence has been found across sectors, ranging from education and journalism to social networking and open-source software development. Social media, collaborative sites, and virtual communities have helped facilitate users from passive consumption to active engagement in creating, curating, and sharing content.

This change requires an interdisciplinary comprehension of the mechanisms underlying digital communities, the socio-technical systems that support them, and the ramifications for privacy, identity, and group behavior. Literature considering these questions ranges across communication theory, computer-supported cooperative work (CSCW), human–computer interaction (HCI), sociology, and media studies. By examining leading contributions from these fields, this research hopes to construct a unified theoretical framework for the examination of technology and human agency in the modern digital landscape.

2 Literature Review

The quick pace of advancement in digital technologies has resulted in a significant shift in the production, dissemination, and consumption of information. The theory of convergence culture, as discussed by Jenkins [1], reflects the blending of old and new media forms, whereby consumers not only watch but also become involved in generating content. This paradigm shift is also seen in educational and institutional media, including campus radio, which has been adjusted to digital environments to facilitate greater accessibility and interactivity [2].

At the same time, the emergence of social media has profoundly altered social dynamics and communication models. Boyd and Ellison [3] describe social networking sites as web-based services that enable individuals to build a public profile, define a list of connections, and examine others' profiles in the system. Kaplan and Haenlein [4] build on this by categorizing social media into different categories including collaborative projects, blogs, content communities, and virtual social worlds and emphasizing both their potential and inherent pitfalls.

Underlying these sites is the process of participatory culture, in which users participate in shared content creation. Benkler [5] describes this as a transition from proprietary to commons-based peer production, challenging conventional models of ownership and control. Shirky [6] supports this argument, highlighting the ability of digital tools to facilitate large-scale, decentralized cooperation without formal organizational forms.

The design of such systems tends to be based on open-source development methodologies. Raymond [7] in the Cathedral and the Bazaar compares disciplined software development with a more community-oriented approach, where openness and peer review drive innovation. Torvalds and Diamond [8] also chronicle the emergence of Linux as a case study in cooperative software engineering. These concepts

are further supported by Loeliger and McCullough [9], who detail how tools such as Git enable version control and distributed collaboration.

Yet, the development of networked communities is not without social and ethical consequences. Acquisti and Gross [10] study privacy issues on sites such as Facebook, discovering a disconnect between users' experience and the revealed visibility of provided information. Baym [11] investigates the ways in which people manage identity, intimacy, and community within online environments, while Putnam [12] compares online social capital with declining traditional civic commitment.

Online communities themselves are designed. Kraut and Resnick [13] advance evidence-based solutions for developing sustainable online communities, including policies of moderation and social feedback mechanisms. Holland and Naudé [14] see that marketing there becomes a problem of information handling, calling for adaptive tools and user analytics. Schmidt and Bannon [15] advance the concept of articulation work in cooperative systems, highlighting the frequently hidden work required to align tasks and workflows.

In the meantime, Rheingold [16] and Nardi and O'Day [17] situate online interaction as being embedded in larger information ecologies. These spaces are defined by the interplay of individuals, practices, technologies, and values. This is consistent with the socio-cultural view provided by Lave and Wenger [18], who explain learning as a process of legitimate peripheral participation in communities of practice.

In addition, researchers have moved to visualization and interaction to gain a deeper insight into user behavior. Viegas and Donath [19] discuss graphical views of chat conversations and their implications for presence and participation. Golder and Huberman [20] study collaborative tagging systems and uncover emergent patterns in how people categorize and access information.

Methodologically, researchers are confronted with difficulties in capturing and analyzing visual social media data. Young [21] considers the need to balance ethical concerns with the requirements of digital ethnography, especially anonymizing and archiving visual material. This is especially relevant in the current media-rich environment, where content lingers beyond its original context.

At the macro level, Castells [22] puts these changes in the context of the network society, where information flow determines economic and social forms. O'Reilly [23] referred to this change as Web 2.0, with features such as user-generated content, interoperability, and collective intelligence.

In aggregate, these publications offer a nuanced basis for seeing the intersection of media, technology, and society. As scholarly work continues, it is vital to synthesize theoretical understanding and empirical approaches in order to design more ethical, equitable, and effective digital technologies.

3 Proposed System

Codeshare aims to address the limitations of existing platforms by introducing a unified platform specifically designed for coders. This platform will integrate the core functionalities of LinkedIn, GitHub, and Instagram, providing a centralized hub for project showcasing, social networking, code sharing, and collaborative learning.

3.1 Project Showcase

Unlike the fragmented project showcasing capabilities of existing platforms, Codeshare will provide a dedicated space for coders to present their work in a visually compelling manner. This will involve incorporating multimedia elements, detailed descriptions, and interactive exploration features to enhance project visibility and facilitate discovery.

3.2 Social Networking

Codeshare will foster a vibrant and interconnected community by integrating social networking features inspired by LinkedIn and Instagram. This will enable coders to connect with peers, engage in discussions, and participate in collaborative activities, fostering a sense of belonging and promoting knowledge sharing.

3.3 Code Sharing and Collabaration

Addressing the limitations of existing platforms in supporting collaborative coding, Codeshare will integrate seamlessly with version control systems like GitHub and GitLab. This will facilitate code sharing, interactive code reviews, and potentially real-time collaborative coding, enhancing teamwork and code quality.

3.4 Q&A and Support

To foster a supportive learning environment, Codeshare will incorporate a dedicated Q&A section. This will enable coders to seek assistance, share knowledge, and engage in discussions, creating a community-driven support system and promoting continuous learning.

3.5 Comparison

See Tables 1 and 2.

Table 1 Proposed versus other models

Feature	Proposed	Other
Project showcase	Visually driven, detailed information, interactive exploration	Fragmented, limited visuals, code-centric
Social networking	Integrated with project showcase and code sharing, dedicated groups, activity feeds	Separate from project/code context, limited features
Code sharing	Integrated version control, code review, real-time collaboration	Limited, lacks social context
Q&A and support	Dedicated forum, voting system, mentorship opportunities	Limited, within groups or repositories
Content feed	Coding-related content sharing, integration with other platforms, curated channels	General or visual focus, not coding-specific
Overall	Unified platform for coders, integrating project showcase, social networking, code sharing, and Q&A	Fragmented, catering to specific needs

Table 2 Existing versus codeshare

Feature	Existing	Codeshare
Primary function	Instagram—social media and content sharing GitHub—code hosting and version control LinkedIn—professional networking	Unified platform for coding, collaboration, networking, and learning
Target users	Instagram—general public GitHub—developers and teams LinkedIn—job seekers and professionals	Developers, students, mentors, and tech communities
Real-time coding collaboration	All platforms are asynchronous or lack native coding tools	Synchronous live coding with collaborative editing
Project discovery and matching	Instagram and GitHub rely on manual search, LinkedIn offers job recommendations	AI-based matching for projects and collaborators
Integrated communication tools	Instagram—DMs, LinkedIn—messaging, GitHub lacks built-in real-time chat	Built-in chat, for instant team interaction
End-to-end collaboration environment	All platforms focus on isolated functions (social, hosting, or networking)	All platforms focus on isolated functions (social, hosting, or networking)

3.6 Novelty

The novelty of our platform lies in the holistic integration of AI across the entire collaboration cycle. It is not just a coding interface or a repository system—it is a smart, interactive, and learning-enabled environment where developers:

- Discover projects and teammates through machine learning-based matching.
- Collaborate in real-time using synchronized editors and communication tools.
- Resolve merge conflicts with AI-powered resolution suggestions.
- Improve code quality through live automated code reviews and feedback.
- Develop skills via AI-suggested learning pathways and mentor recommendations.

This unified approach has not been implemented in existing platforms, making this research a first step towards truly intelligent and community-focused development environments.

3.7 Related Work

The rise of collaborative development platforms has brought significant changes to how software is built and shared. Popular platforms like GitHub, GitLab, and Bitbucket have streamlined version control, issue tracking, and asynchronous collaboration. However, these platforms still rely heavily on manual project search, delayed feedback loops, and minimal support for real-time engagement or learning-focused collaboration.

Boyd and Ellison [3] laid the groundwork for understanding social networking structures, highlighting their role in enabling collaboration and user-driven communities. Similarly, Jenkins [1] and Shirky [6] discussed how participatory culture and decentralized organization can empower users to co-create and innovate. However, these works focus on broader social media dynamics, lacking technical implementation specific to coding environments.

Recent efforts like Visual Studio Live Share have introduced real-time coding capabilities, but they often operate as plugins without deep integration into version control, project discovery, or AI-based feedback systems. Stack Overflow, on the other hand, offers community-based Q&A, but lacks the contextual awareness and personalization necessary for dynamic peer learning or project recommendations.

More recently, AI has been applied in platforms like GitHub Copilot, which uses transformer-based models to assist in code generation. While powerful, Copilot is an individual programming aid rather than a full-fledged collaborative and community-driven platform. It lacks live team collaboration, recommendation systems, or automated conflict resolution for multi-user projects.

4 System Architecture

To support the features and functionalities described in the proposed system, Codeshare will employ a robust and scalable system architecture. The architecture will be based on a microservices approach, allowing for independent development, deployment, and scaling of individual components. The key components of the system architecture are as follows.

4.1 Client-Side

User Interface (UI): The UI will be developed using React, a popular JavaScript library for building dynamic and interactive user interfaces.

State Management: State management will be managed using a library like Redux or Zustand to ensure efficient data flow and updates within the UI.

API Interaction: The UI will interact with the backend microservices through API calls using libraries like Axios or Fetch API.

4.2 API Gateway

Centralized Entry Point: An API gateway will serve as a single-entry point for all API requests, providing a unified interface for clients to interact with the backend services.

Authentication and Authorization: The API gateway will handle user authentication and authorization, ensuring that only authorized users can access specific resources.

Routing and Load Balancing: The API gateway will route requests to the appropriate microservices and distribute traffic across multiple instances of each service to ensure high availability and performance.

4.3 Microservices

Modular Functionality: Core functionalities of Codeshare will be implemented as microservices, including:

User Service: Manages user accounts, profiles, and social connections.

Project Service: Handles project creation, editing, searching, and display.

Code Service: Manages code repositories, version control, and collaboration.

Content Service: Handles the content feed, including posting, sharing, and curating content.

Notification Service: Manages real-time notifications for new messages, comments, and other events.

Search Service: Provides search functionality across users, projects, and content.

4.4 Data Store

MongoDB: MongoDB will be used as the primary database to store user data, project data, code metadata, content, and other platform information. Its flexible schema and ability to handle unstructured data make it suitable for the diverse data types within Codeshare.

4.5 External Services

GitHub/GitLab: Codeshare will integrate with GitHub and GitLab to leverage their code repository management and version control capabilities.

AWS S3: AWS S3 will be used for storing images, videos, and other media files associated with projects and content.

4.6 Architectural Diagram

The architectural diagram provides a visual representation of the system's structure and the interactions between its various components, including clients, API gateway, microservices, databases, and external services. It illustrates the flow of data and requests through the system, highlighting the modular design and the integration of different technologies to support the platform's functionalities and scalability (Fig. 1).

A. Architecture Components of Codeshare

1. Clients
   Users access the system using web browsers or mobile apps.
   Clients send requests to the application through the Load Balancer.

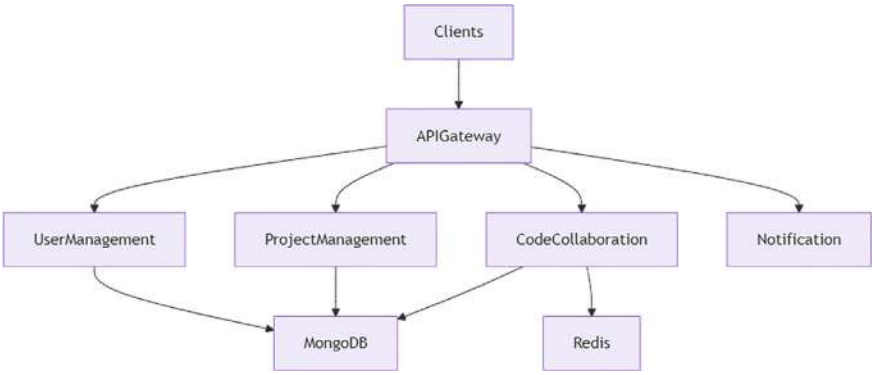


Fig. 1 Architecture diagram

2. Load Balancer

   Distributes incoming client requests across multiple servers to ensure high availability and scalability.

   It routes client requests to the API Gateway.

3. API Gateway

   - The API Gateway is responsible for handling all incoming API requests from clients.
   - It manages:
     - Authentication and Authorization of users.
     - Routing requests to the appropriate microservice.
     - Rate Limiting to prevent abuse of services.
   - After processing, requests are sent from the API Gateway to appropriate Microservices.

4. Microservices

   - The application has several microservices that each serve a distinct purpose:

     User Management Service: Handles user registration, login, profile management, etc.

     Project Management Service: Manages project creation, editing, and tracking.

     Code Collaboration Service: Manages code collaboration features between users.

     Content Feed Service: Provides the content feed that users interact with.

     Notification Service: Handles real-time notifications and alerts.

     Search Service: Manages search functionality across projects, users, and content.

- Each microservice is independently deployable and communicates through internal REST APIs or message queues.

5. Database (MongoDB)

- MongoDB is used for data storage, including:
  User Data: Personal details, account information, preferences.
  Project Data: Information related to projects, like metadata and statuses.
  Content: Posts, comments, likes, etc.
- All Microservices that need to store or retrieve data interact with MongoDB.

6. Cache (Redis)
   Redis is used to cache frequently accessed data to speed up the performance.
   Microservices interact with Redis to retrieve cached data before querying MongoDB.
7. External Services

- GitHub/GitLab Integration
  The Code Collaboration Service integrates with GitHub and GitLab to allow users to link and manage code repositories.
- AWS S3
  Used for storing large files and assets, such as user-uploaded images, project documentation, etc.
  The Project Management Service and User Management Service interact with AWS S3 for storing and retrieving files.

B. Interactions Flow

- Client Request Flow
  Clients (web or mobile) send a request to the Load Balancer.
  The Load Balancer forwards the request to the API Gateway.
- API Gateway Routing
  The API Gateway authenticates the client request.
  It then routes the request to the appropriate Microservice based on the request type.
- Microservice Actions
  The chosen Microservice performs the required operations.
  If data retrieval or storage is required, the Microservice communicates with MongoDB.
  If the data is frequently accessed, Redis is checked first to reduce load times.
- External Integrations
  When project repositories need to be linked, the Code Collaboration Service interacts with GitHub/GitLab.

If users upload files, the Microservice interacts with AWS S3 for file storage.

5 Project Design and Features

The key features of Codeshare are designed to cater to the multifaceted needs of coders, offering a comprehensive platform that merges social networking, coding, and learning opportunities.

5.1 Project Showcase

The Project Showcase feature of Codeshare will allow users to present their projects in an engaging, visual manner. It is designed to be a space where coders can display their work with the same ease as they would on GitHub, but with more visually appealing presentation options. This feature aims to create a portfolio-like experience where each project is a showcase of the coder's skills and creativity.

- Project Presentation: Coders can upload detailed descriptions of their projects, complete with multimedia support, including images, videos, and demo clips. These elements can be used to illustrate the project's functionality, design choices, and user experience, offering potential collaborators or employers a thorough understanding of the project briefly. Each project will also allow users to include links to external repositories, such as GitHub or GitLab, providing direct access to the codebase and documentation.
- Search and Filtering: The platform will have advanced search capabilities, enabling users to filter and discover projects based on specific criteria. Coders can search by programming languages (e.g., Python, JavaScript), technologies (e.g., React, Django), or categories such as mobile development, web development, data science, machine learning, and more. This makes it easier for users to find inspiration, follow trends, or identify potential collaborators who are working on similar projects.
- Featured Projects Section: To further highlight outstanding work, Codeshare will have a "Featured Projects" section. This feature will showcase projects that stand out in terms of innovation, design, or technical complexity. Projects can be featured through a community voting system, where users can upvote projects they find impressive, or they may be curated by platform moderators based on certain criteria, such as community engagement or technical merit. This section provides users with an opportunity to have their work promoted to a broader audience, potentially attracting more feedback, collaboration opportunities, or even job offers.

5.2 Social Networking

The **Social Networking** component of Codeshare will enable users to build and expand their professional coding network. The design of this feature is inspired by the professional focus of LinkedIn, combined with the engagement and interaction style of Instagram.

- User Profiles: Each coder on Codeshare will be able to create a comprehensive profile that showcases their skills, experience, education, and completed projects. Profiles will serve as a combination of a resume and portfolio, where users can highlight their technical expertise, certifications, and achievements. Additionally, profiles can include links to external platforms such as GitHub, LinkedIn, and personal websites.
- Connections and Groups: Users can connect with other coders to expand their professional network. The platform will facilitate one-on-one connections, allowing users to send invitations to collaborate, ask for advice, or simply follow each other's work. Additionally, the platform will support groups, where users can join coding communities focused on languages, frameworks, or fields (e.g., Python Enthusiasts, JavaScript Frameworks, AI and Machine Learning). These groups will enable discussions, project collaborations, and knowledge sharing among members with similar interests.
- Activity Feed and Engagement: Similar to Instagram's social engagement model, users on Codeshare will have access to a live activity feed. The feed will display updates from connections and groups, including new projects, shared content, and coding tips. Users can like, comment on, and share posts, fostering a sense of community and dialogue around coding topics. This will encourage interaction and collaboration, making the platform more dynamic and engaging.
- Group Functionality: Codeshare's group functionality is designed to facilitate community-driven collaboration. Users can form groups around shared interests, such as a particular programming language or an ongoing project. Groups can serve as hubs for discussions, sharing resources, organizing hackathons, or working on open-source projects. Group interactions will help coders connect with like-minded individuals, foster mentorship relationships, and expand their network within their niche areas of expertise.

5.3 Code Sharing and Collaboration

Codeshare's **Code Sharing and Collaboration** features are designed to streamline the process of working together on coding projects in real-time, making it easier to engage in collaborative coding and peer review, especially for open-source and team-based projects.

- GitHub and GitLab Integration: To enable seamless collaboration, Codeshare will integrate directly with GitHub and GitLab, two of the most popular platforms for

version control and collaborative development. This integration will allow users to link their repositories directly to their profiles, making it easy for others to explore their codebase, contribute to projects, or provide feedback.

- Real-Time Code Collaboration: Codeshare will provide a built-in, real-time code editor similar to Google Docs but tailored for coding. This editor will allow multiple users to simultaneously work on the same project, with live updates and collaborative coding features. For example, team members working on the same project can contribute code in real time, see each other's changes as they happen, and provide feedback instantly.
- Live Code Reviews: The platform will include functionality for live code reviews, where users can leave comments, suggest edits, and approve changes directly within the code editor. This process will enhance the collaborative aspect of project development, enabling users to quickly iterate on their work based on peer feedback. The ability to review and discuss code in real time makes this feature particularly useful for open-source projects and hackathons, where rapid feedback and iterative improvement are crucial.
- Discussion and Suggestions: Codeshare's collaboration tools will also allow users to engage in discussions and provide suggestions within project workspaces. Users can propose new features, identify potential bugs, or discuss improvements, all within the context of a shared coding environment. This collaborative approach to coding helps build stronger teams, encourages learning, and improves the overall quality of the code.

5.4 Q&A and Help

Codeshare will foster knowledge sharing and learning through its **Q&A and Mentorship** features, providing coders with opportunities to ask questions, share knowledge, and receive guidance from experienced professionals.

- Q&A Forum: Modelled after the popular Stack Overflow, the platform will include a dedicated Q&A forum where users can post technical questions related to coding, algorithms, frameworks, debugging, and more. Other users can respond with answers, and the community can upvote the best responses. The voting system will help ensure that the most helpful and accurate answers are easily visible, fostering a positive and supportive community of knowledge sharing.
- Reputation Points: Encourage high-quality contributions, users will earn reputation points for answering questions, providing useful feedback, and participating in discussions. These reputation points will serve as a form of recognition within the community, highlighting a user's expertise and dedication to helping others. High-reputation users will stand out to potential collaborators or employers, making this an additional incentive for coders to be active in the community.
- Mentorship Programs: Codeshare will feature a mentorship program designed to connect experienced developers with novices. This program will enable

less experienced coders to seek guidance, advice, and career mentorship from seasoned professionals. Mentorship can take many forms, including project-based mentoring, where mentors guide mentees through a specific project, or career-focused mentoring, where the focus is on helping mentees navigate job opportunities, technical interviews, and skill development.

- AMA (Ask Me Anything) Sessions: To further engage the community, Codeshare will regularly host AMA (Ask Me Anything) sessions with prominent figures from the coding world, such as experienced software engineers, project managers, or technology leaders. These sessions will provide users with the opportunity to ask questions related to industry trends, career advice, technical skills, and the future of software development. This feature will give coders direct access to insights and advice from leaders in the field.

5.5 Content Feed

The **Content Feed** will serve as the main hub for coders to share and engage with a variety of coding-related content, from blog posts to tutorials and articles.

- Sharing Content: Users will be able to share blog posts, coding tutorials, technical articles, videos, and other relevant content. The platform will also integrate with external platforms like Medium and YouTube, allowing users to share content they've created on these platforms directly to their Codeshare profile.
- Personalized Recommendations: The content feed will be personalized for each user based on their interests, preferences, and activity. This algorithm-based feed will recommend relevant blog posts, articles, tutorials, coding challenges, and events tailored to each user's unique needs. For example, a user who frequently participates in machine learning discussions may see more articles and challenges related to that topic in their feed.
- Content Engagement: Users will be able to engage with shared content by liking, commenting, and sharing it with their network. This interactive content feed will not only provide a platform for coders to showcase their knowledge but will also foster discussions and feedback, encouraging knowledge sharing and peer learning.

6 Technology Stack

Codeshare will be developed using modern, scalable technologies that allow for a smooth user experience and efficient backend performance. The primary components include.

6.1 Frontend

The frontend will be built using **React**, a popular JavaScript library known for its efficiency in handling dynamic user interfaces. React's component-based architecture makes it easy to manage and update specific elements of the platform without affecting the entire application. This is particularly important for a platform like Codeshare, where users interact with real-time elements such as live feeds, collaborative coding tools, and project showcases. React's extensive community and ecosystem ensure ongoing support and updates, making it a reliable choice for the platform's frontend.

6.2 Backend

The backend will be powered by **Node.js** with **Express**, providing a fast and scalable environment for handling requests. **Node.js** is widely recognized for its ability to manage multiple simultaneous connections, making it ideal for real-time collaboration tools. The **Express** framework will streamline the process of building APIs and managing the platform's core logic, ensuring that the backend is both robust and flexible.

6.3 Database

MongoDB, a NoSQL database will serve as the database for Codeshare. As a NoSQL database, MongoDB is well-suited for handling unstructured or semi-structured data, making it ideal for storing user profiles, project data, and posts. Its scalability ensures that Codeshare can grow with its user base, accommodating large volumes of user-generated content without sacrificing performance. MongoDB's document-oriented structure also allows for flexible data modeling, which is essential when dealing with complex entities like projects and collaborative workspaces.

6.4 Cloud Hosting

AWS (Amazon Web Services) will provide the cloud infrastructure for Codeshare, ensuring that the platform remains reliable and scalable as its user base expands. AWS services such as **EC2** for compute power and **S3** for storage will be used to host the platform and store user-generated content, respectively. AWS's global presence will allow Codeshare to offer fast, localized services to users around the world, while its robust security measures will ensure that user data remains safe.

7 Results and Discussion

The development and implementation of the Codeshare platform were rigorously evaluated through a series of experiments and user interactions to assess its effectiveness, efficiency, and usability. This section presents the quantitative and qualitative outcomes of these evaluations, highlighting the system's performance across various functionalities and its impact on the user experience. Below are the results of the Codeshare platform implemented as a Streamlit Web Application.

Register Page (Fig. 2): This figure is used for users to create a new account by signing up using email ID and password, or by signing in with Google or Facebook.

Login Page (Fig. 3): This figure is used to allow users to log in to their account by providing their email ID and password.

Forgot Password Popup (Fig. 4): This figure is used for sending reset password links to the user's email in case they forget their login credentials.

Forgot Password Popup (Fig. 5): This figure is used for sending reset password links to the user's email in case they forget their login credentials.

Home/Dashboard (Fig. 6): This figure is used to show the projects uploaded by the users, along with options for interacting with them (like, share, etc.).

The development of Codeshare is driven by the need for a unified platform that addresses the limitations of existing platforms for coders. The results of the research indicate a strong demand for a platform that integrates project showcasing, social networking, code sharing, and Q&A support within a dedicated community. The research revealed that coders often use multiple platforms to fulfil their diverse needs. Project showcasing is often fragmented across platforms like GitHub and Instagram,

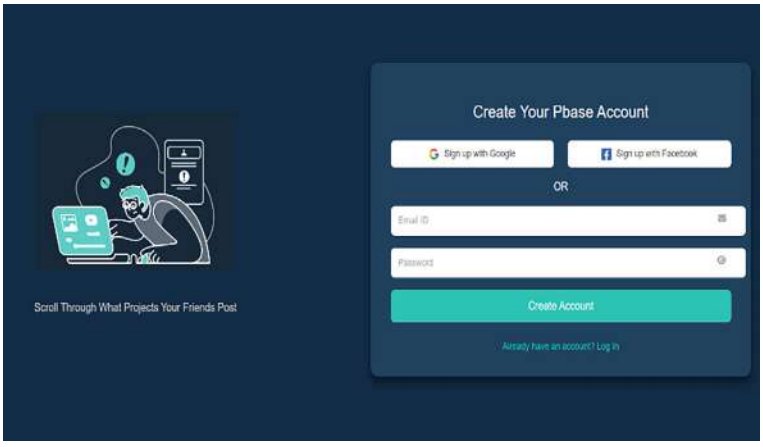


Fig. 2 Registration page

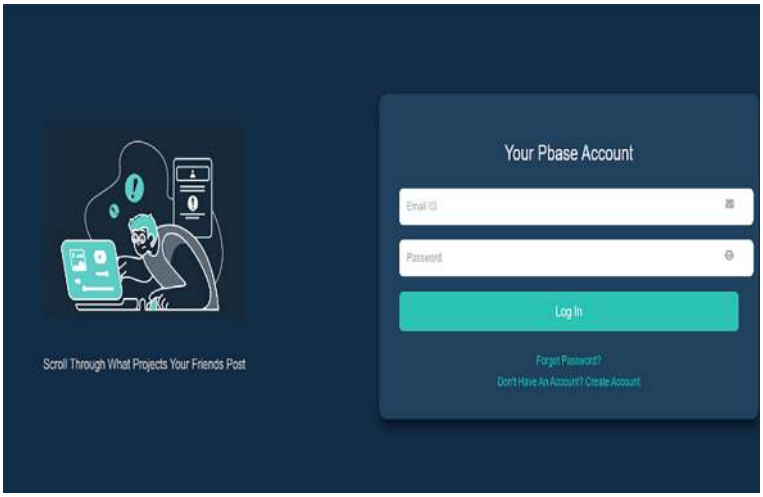


Fig. 3 Login page

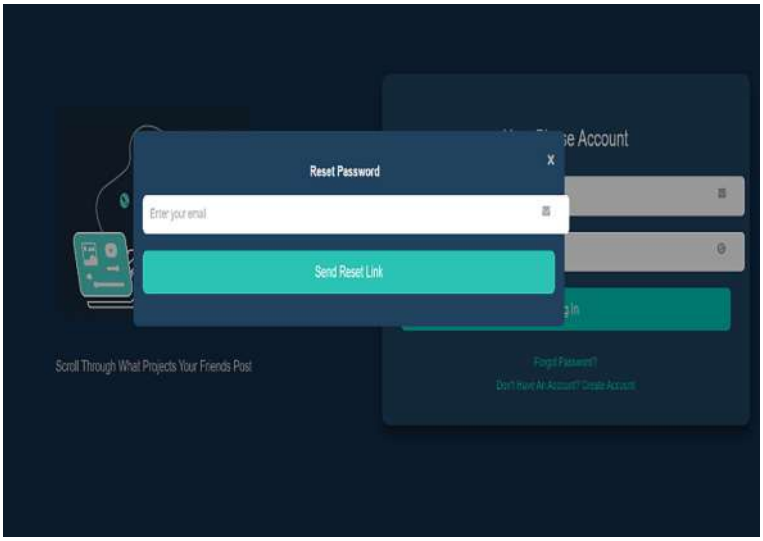


Fig. 4 Forgot password

while social networking and community engagement are primarily relegated to platforms like LinkedIn. This fragmentation hinders effective knowledge sharing and collaboration. Codeshare addresses these challenges by providing a unified platform that integrates the core functionalities of existing platforms. The proposed system offers a visually driven project showcase, dedicated social networking features, integrated code sharing and collaboration tools, and a dedicated Q&A forum. This

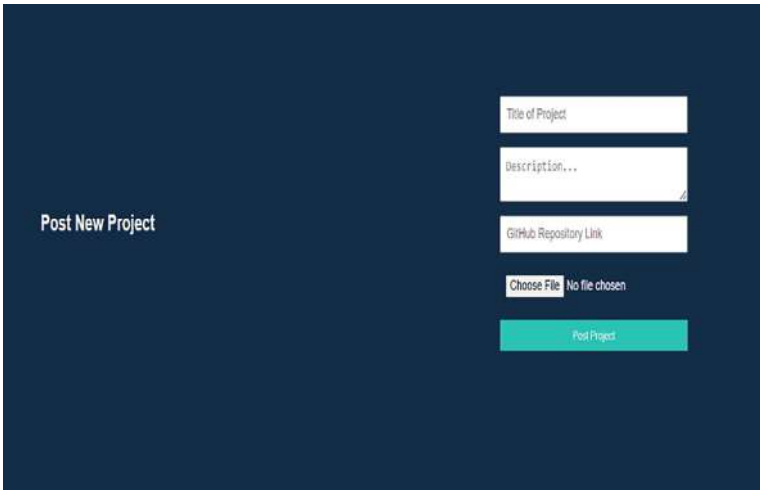


Fig. 5 Upload project

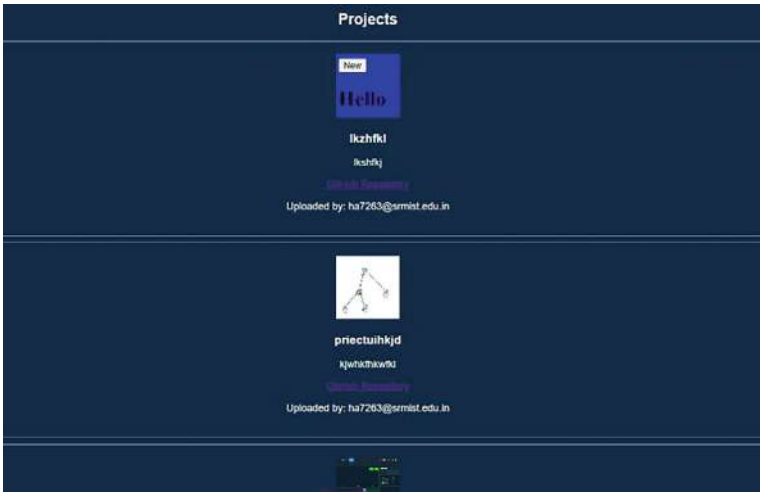


Fig. 6 Dashboard

integrated approach aims to foster a more interconnected and collaborative coder community. The visually driven project showcase will enable coders to present their work in a more engaging and accessible manner, increasing visibility and recognition. The integrated social networking and code sharing features will promote collaboration and knowledge sharing among coders. dedicated Q&A forum and mentorship opportunities will create a supportive learning environment for coders of all levels.

The unified platform will streamline coders' workflow by providing a centralized hub for their diverse needs.

8 Conclusion

Codeshare has the potential to revolutionize the way coders connect, collaborate, and showcase their work. By merging the functionalities of LinkedIn, GitHub, and Instagram, it offers a unique, all-encompassing platform that caters to the diverse needs of the coder community. Whether it's through project showcasing, social networking, real-time collaboration, or knowledge sharing, Codeshare provides coders with a space where they can build meaningful connections, advance their careers, and contribute to the wider coding ecosystem. With its focus on user experience, scalability, and security, Codeshare is poised to become a central hub for coders worldwide. The proposed AI-powered coding platform introduces a comprehensive solution to key limitations found in existing collaborative development environments. By integrating AI-based project and peer recommendations, real-time collaboration capabilities, and intelligent merge conflict resolution, this platform improves both the efficiency and quality of software development. It ensures developers are matched with relevant projects and teammates, enables seamless live interaction, and reduces manual workload through smart automation. The main contribution of this paper is the development of a multi-functional, AI-driven collaborative ecosystem that supports end-to-end project development—from discovery and coding to review and deployment. Unlike traditional platforms, it emphasizes not only collaboration but also learning, security, and productivity, making it a valuable tool for both novice and experienced developers. This research sets the groundwork for future innovations in collaborative software engineering by demonstrating how artificial intelligence can be meaningfully integrated into the development lifecycle to foster more intelligent, efficient, and inclusive programming communities.

Acknowledgements We would like to express our gratitude to the authors of the research papers used in this study for their valuable contributions to the field.

References

1. Jenkins, H.: Convergence Culture: Where Old and New Media Collide. NYU Press (2006)
2. Fajardo, K.B.G.: Campus radio version 2.0: the convergence of campus radio with digital media. Talastásan Philippine J. Commun. Media Stud. **1**(1), 71–84 (2022)
3. Boyd, D.M., Ellison, N.B.: Social network sites: definition, history, and scholarship. J. Comput.-Mediat. Commun. **13**(1), 210–230 (2007)
4. Kaplan, A.M., Haenlein, M.: Users of the world, unite! The challenges and opportunities of social media. Bus. Horiz. **53**(1), 59–68 (2010)

5. Benkler, Y.: The Wealth of Networks: How Social Production Transforms Markets and Freedom. Yale University Press (2006)
6. Shirky, C.: Here Comes Everybody: The Power of Organizing Without Organizations. Penguin Books (2008)
7. Raymond, E.S.: The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary. O'Reilly Media (1999)
8. Torvalds, L., Diamond, D.: Just for Fun: The Story of an Accidental Revolutionary. HarperCollins (2001)
9. Loeliger, J., McCullough, M.: Version Control with Git, 2nd edn. O'Reilly Media, Inc. (2012)
10. Acquisti, A., Gross, R.: Imagined communities: awareness, information sharing, and privacy on the Facebook. In: Privacy Enhancing Technologies Symposium (PETS) (2006)
11. Baym, N.K.: Personal Connections in the Digital Age. Polity Press (2010)
12. Putnam, R.D.: Bowling Alone: The Collapse and Revival of American Community. Simon & Schuster (2000)
13. Kraut, R.E., Resnick, P.: Building Successful Online Communities: Evidence-Based Social Design. MIT Press (2012)
14. Holland, C.P., Naudé, P.: The metamorphosis of marketing into an information handling problem. J. Bus. Ind. Mark. **19**(3), 167–177 (2004)
15. Schmidt, R., Bannon, L.: Taking CSCW seriously: supporting articulation work. Comput. Supported Coop. Work (CSCW) **1**(1–2), 7–40 (1992)
16. Rheingold, H.: The Virtual Community: Homesteading on the Electronic Frontier. MIT Press (2000)
17. Nardi, B.A., O'Day, V.L.: Information Ecologies: Using Technology with Heart. MIT Press (1999)
18. Lave, J., Wenger, E.: Situated Learning: Legitimate Peripheral Participation. Cambridge University Press (1991)
19. Viegas, F.B., Donath, J.: Chat circles. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 9–16 (1999)
20. Golder, S.A., Huberman, B.A.: Usage patterns of collaborative tagging systems. J. Inf. Sci. **32**(2), 198–208 (2006)
21. Young, A.M.: Approaching instagram data: reflections on accessing, archiving and anonymising visual social media (2020)
22. Castells, M.: The Rise of the Network Society: The Information Age: Economy, Society, and Culture. Wiley-Blackwell (2010)
23. O'Reilly, T.: What Is Web 2.0? Design Patterns and Business Models for the Next Generation of Software (2005)

Spatio-Temporal Land Use and Land Cover Analysis and Urban Expansion Prediction Using Remote Sensing and SMOTE-SVM Classification

Priya Surana, Pramod Patil, and Baravkar Shruti

Abstract This study investigates spatio-temporal land use and land cover (LULC) changes in Pune using Landsat 8 imagery and advanced machine learning techniques. Google Earth Engine was employed to classify imagery from 2015 to 2023 into six LULC categories, with a SMOTE-enhanced Support Vector Machine (SVM) model addressing class imbalance and achieving high classification accuracy. Outputs were analyzed in QGIS to detect urban expansion trends, revealing substantial growth in built-up and road areas, coupled with notable declines in agricultural and forest zones. A regression-based forecasting model was developed in Python to predict LULC patterns for 2026. Results indicate continued urban intensification and ecological contraction. The study provides a scalable geospatial framework for sustainable urban land management and informs evidence-based planning and policy formulation.

Keywords Land use and land cover (LULC) · Remote sensing · SMOTE-SVM classification · Urban expansion · Predictive modelling · Google Earth Engine (GEE)

1 Introduction

Urbanization stands as one of the most transformative forces of the twenty-first century, reshaping economic structures, ecological systems, and socio-spatial configurations globally. Rapid demographic transitions—largely driven by rural-to-urban migration—have resulted in significant alterations to land use and land cover (LULC), particularly in rapidly expanding metropolitan regions. While urban growth catalyzes economic activity and infrastructural development, it concurrently triggers

P. Surana (✉) · B. Shruti
Computer Engineering, Pimpri Chinchwad College of Engineering, Pune, India
e-mail: priya.surana@pccoepune.org

P. Patil
Computer Engineering, D.Y. Patil Institute of Technology, Pune, India

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_27

systemic challenges including land degradation, resource scarcity, and environmental disequilibrium [1].

The Pune metropolitan region, located in Maharashtra, India, exemplifies these urban pressures. Renowned for its educational, technological, and industrial prominence, Pune has experienced sustained population influxes and spatial expansion in recent decades. These dynamics have led to unregulated urban sprawl, depletion of agricultural land, and fragmentation of ecological corridors. There is, therefore, an urgent requirement for scalable, high-resolution techniques capable of monitoring LULC transitions and predicting future land use trajectories [2, 3].

Traditional cartographic methods lack the spatial and temporal resolution required for proactive urban planning. In response, this study harnesses Earth observation data and advanced machine learning—specifically a SMOTE-enhanced Random Forest classifier—to conduct a multi-year LULC classification and forecast urban growth patterns using linear regression [4, 5]. The efficacy of such classifiers has been demonstrated in prior studies, especially in environments with heterogeneous landscapes and class imbalance challenges [1, 6, 7].

The primary contribution of this research lies in its integrative framework that combines temporal satellite imagery analysis, imbalance correction, and predictive modelling to generate actionable insights for sustainable urban planning. Unlike prior works that focus primarily on classification or temporal analysis in isolation [8, 9], this study bridges both diagnostic and prognostic perspectives using Google Earth Engine, QGIS, and Python-driven regression techniques.

Through a detailed analysis of spatial patterns from 2015 to 2023 and forecasting scenarios for 2026, this study offers a replicable, data-driven approach to urban land monitoring. The work contributes not only methodologically by refining classification and prediction strategies for imbalanced geospatial data [10, 11], but also practically by informing planning interventions in one of India's fastest-growing urban environments.

The principal novelty of this research lies in the combined application of SMOTE-enhanced SVM classification on multi-temporal medium-resolution imagery with integrated predictive regression modeling—thereby unifying diagnostic mapping and prognostic forecasting within a single framework.

2 Material and Methods

2.1 Related Work

Carranza-García et al. [8] developed a generalized evaluation framework for Convolutional Neural Network (CNN)-based LULC classification using five benchmark datasets: three hyperspectral (Indian Pines, Pavia University, Salinas) and two radar (San Francisco, Flevoland). Their uniform 5 × 5-patch CNN, trained with extensive data augmentation and 5 × 3-fold cross-validation, achieved exceptional overall

accuracies (96.78–99.36%), outperforming SVM, Random Forest, and kNN baselines. The study's strength lies in its statistically robust, reproducible protocol and its demonstration of CNN resilience to class imbalance via dropout regularization. However, its static-image focus precludes temporal dynamics, and the generic network architecture is not optimized for dataset-specific nuances. Our work extends this paradigm by integrating imbalance correction (SMOTE) with Random Forest on multi-temporal Landsat data and embedding predictive modeling to forecast future LULC trends in Pune.

Yang et al. [12] investigated LULC classification using high-resolution (20 cm) aerial imagery and nDSM data over Hameln, Germany, deploying two CNN strategies: SegNet variants for land cover (LC) and a lightweight patch-based LiteNet for land use (LU). Their best SegNet ensemble (EN(B0, B1, O, F)) achieved 85.7% overall accuracy (mean F1 = 76.6%), while the LiteNet ensemble reached 77.4% OA (mean F1 = 63.1%). Effective ensemble strategies and transfer learning (CamVid weights) bolstered model generalization, and custom patch sampling preserved object geometry. However, performance dipped on underrepresented classes, and deeper SegNet models exhibited marginal gains, indicating overfitting risk. In contrast, our SMOTE-SVM approach directly addresses minority-class imbalance, offering improved class separability across heterogeneous urban landscapes and facilitating temporal change detection beyond static, high-resolution analyses.

Xie and Huang [9] proposed a hybrid pattern-recognition scheme combining Fuzzy C-Means clustering, Support Vector Machine (SVM), and Ensemble Extreme Learning Machine (ELM) to classify high-resolution (2.5 m) satellite imagery of Yuhuatai District (2018–2019), and benchmarked on hyperspectral datasets (Indian Pines, PaviaU, Salinas). Their approach yielded robust accuracies—Indian Pines OA = 86.2% (Kappa = 0.84) and PaviaU OA > 91% (Kappa = 0.88)—demonstrating resilience to spectral noise and reduced sensitivity to outliers. The fusion of spectral, spatial, and textural features enhanced class discrimination. Limitations include generalized dependency on Euclidean metrics and restricted sensor diversity, which may hamper transferability. Our methodology parallels this hybridization ethos by fusing SMOTE oversampling with SVM to rectify imbalance in multispectral Landsat data and extends it with temporal forecasting, thereby strengthening predictive capability across sensor modalities and time series.

Yang et al. [12] evaluated multi-modal CNNs on high-resolution RGB + IR aerial and normalized DSM imagery over two German sites (Hameln, Schleswig) and the ISPRS Vaihingen benchmark. Their SkipNet variants with learnable skip-connections (SkipNet1: IR + height) achieved Hameln OA = 89.6% (mean F1 = 83.2%), while FuseEnc networks recorded OA = 87.3% (Schleswig) and 90.7% (Vaihingen). For LU, their dual-branch LuNet ensembles delivered Hameln OA = 81.7% and Schleswig OA = 78.0%. Strengths include precise boundary delineation via encoder–decoder architectures and effective fusion of spectral and elevation modalities. Yet, small or underrepresented classes remain challenging, and fine-scale object delineation requires further refinement. Comparatively, our SMOTE-SVM framework achieves competitive class balance on lower-resolution Landsat imagery,

demonstrating scalability and temporal extensibility without specialized elevation data.

Yang et al. [13] demonstrated that semantic land cover outputs can enhance land use classification by employing SegNet-based ensembles (EN(B0, B1, O, F): OA = 85.7%, mean F1 = 76.6%) and LiteNet patches (EN(B0, B1): OA = 77.4%, mean F1 = 63.1%) on the same Hameln and Vaihingen datasets. Their rigorous cross-validation and novel patch generation preserved object geometry, improving LU accuracy. However, deeper models offered diminishing returns, and rare classes remained problematic due to sample scarcity. Our work diverges by integrating SMOTE to synthetically augment minority classes within multi-temporal Landsat data, thereby obviating the need for bespoke patches and achieving balanced classification performance across more extensive urban and peri-urban areas, subsequently leveraged for forecasting LULC change.

2.2 Study Area

Ethiopia (Comparative Analysis): Ethiopia, located in the northeastern region of the African continent—commonly referred to as the Horn of Africa—exemplifies diverse physiographic characteristics, ranging from rugged mountain terrains to expansive lowland plains. This topographical heterogeneity fosters an array of ecological zones, rendering the country one of the most biodiverse in sub-Saharan Africa. The Ethiopian highlands are characterized by dense montane forests that serve as vital carbon sinks and biodiversity reservoirs, while the lowlands support extensive agricultural systems that form the backbone of the nation's predominantly agrarian economy. Staple crops such as teff, maize, and coffee dominate the cultivated landscapes, with agricultural productivity closely tied to the seasonal rainfall patterns and altitudinal gradients of the region.

The spatial diversity and ecological variability of Ethiopia make it an ideal case for algorithm benchmarking in land use and land cover classification (Fig. 1). Consequently, this region was selected as a reference area for algorithmic evaluation prior to model transfer and application over the Pune region.

Pune, India (Primary Study Area): The Pune metropolitan region, situated in the western Indian state of Maharashtra, exhibits a complex mosaic of urban, peri-urban, and rural typologies. Geographically positioned between the Western Ghats and the Deccan Plateau, Pune benefits from both climatic diversity and rich geological formations. Historically celebrated as the "Oxford of the East" due to its concentration of premier educational institutions, the city has metamorphosed into a thriving hub for industrial, technological, and cultural enterprises. The urban fabric is marked by high-density residential sectors, commercial enclaves, institutional campuses, and sprawling industrial zones, all of which are visible across the satellite-derived spatial imagery utilized in this study (Fig. 2).

Fig. 1 Study area, Ethiopia

Fig. 2 Analysis and prediction area, Pune

Beyond the urban core, the peri-urban hinterlands retain a blend of agricultural fields, orchards, and open spaces—traces of Pune's agrarian heritage. However, this delicate urban–rural equilibrium is increasingly under threat due to rampant urban sprawl, infrastructural stress, and ecological degradation. Issues such as diminishing green cover, contamination of water resources, vehicular emissions, and irregular land conversion patterns underscore the urgent need for systematic land monitoring frameworks. The influx of migrant populations, drawn by economic and educational prospects, further intensifies developmental pressures on the region's limited land

resources. In response, several sustainability-oriented initiatives—such as the adoption of green building codes, smart city planning frameworks, and integrated mobility networks—have been launched to reconcile urban growth with ecological conservation. Thus, Pune serves not only as the focal geography of this investigation but also as a microcosmic representation of India's broader urbanization trajectory, encapsulating the multifaceted tensions between modernization, ecological preservation, and socio-economic inclusivity.

2.3 Methodology

2.3.1 Data Sources

Ethiopia Dataset (Comparative Baseline): To benchmark the classification algorithms prior to their application on the Pune region, a representative dataset from Ethiopia was utilized. This dataset comprises Tier 1 raw scenes sourced from the USGS Landsat 8 Collection 2 archive, specifically from the Image Collection ee.ImageCollection("LANDSAT/LC08/C02/T1"). These Tier 1 datasets include calibrated digital number (DN) values that represent top-of-atmosphere radiance and are processed to Level-1 Precision Terrain (L1TP) standards. Such data offers high radiometric fidelity and geometric accuracy, making it optimal for temporal LULC assessments.

Landsat 8 imagery was selected for five target years—2015, 2017, 2020, 2021, and 2023—owing to its multispectral capabilities and proven consistency across sensors. Given its well-documented temporal resolution and long-standing reliability in Earth observation, Landsat data remains foundational for robust land cover change detection studies.

Pune Dataset (Primary Study Region): In contrast to Ethiopia, Pune's datasets were not directly available in preprocessed collections on Google Earth Engine (GEE). Therefore, custom Region of Interest (ROI) boundaries were delineated manually across the target years—2015, 2017, 2020, 2021, and 2023—by selecting analogous geographic coordinates for each temporal snapshot. Using high-resolution satellite imagery available on GEE, ROI bounding boxes were defined and applied uniformly across all timeframes to ensure spatial consistency.

Subsequently, representative training samples were extracted for distinct land cover classes including water bodies, road networks, built-up areas, vegetation, forests, and barren land (Figs. 3, 4, 5, 6, 7 and 8). These training samples served as the basis for classification and model calibration.

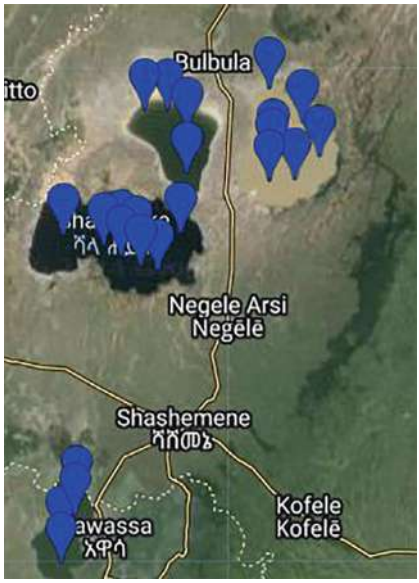


Fig. 3 Representative samples of water bodies

Fig. 4 Road infrastructure samples

Fig. 5 Built-up and urban area samples (Aerial View)

Fig. 6 Built-up and urban area samples (Geographical View)

Fig. 7 Forest cover samples

2.3.2 Image Preprocessing

To ensure temporal coherence in classification, images corresponding to each target year were selectively filtered from the Landsat collections. This filtering process excluded scenes with high cloud cover or anomalies and retained those best suited for time-series analysis.

Following filtration, a compositing technique was employed to generate a single representative image per year. The composite was typically derived using either a median pixel value approach or by selecting imagery with minimal cloud interference. This approach reduces atmospheric noise and captures generalized surface reflectance characteristics.

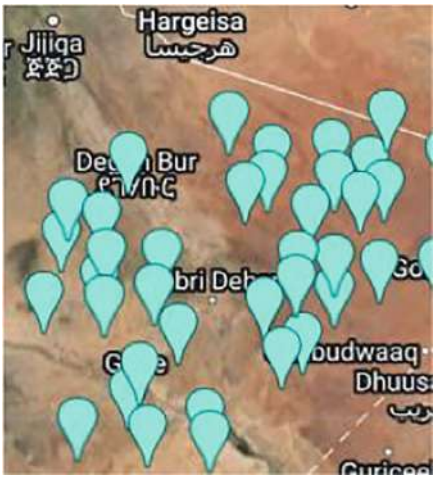


Fig. 8 Barren land samples

Subsequent preprocessing involved the selection of specific spectral bands relevant to LULC classification. These bands, corresponding to different wavelengths of reflected electromagnetic radiation, offer diagnostic insights into various land surface features. For instance, near-infrared and shortwave-infrared bands enhance the differentiation between vegetated and non-vegetated areas, while visible bands aid in discriminating urban structures and water bodies. Optimal band combinations were determined through iterative experimentation to maximize classification accuracy.

2.3.3 Image Classification

Training Phase

We selected the SMOTE-SVM hybrid model because SMOTE effectively balances minority classes in imbalanced LULC datasets, while SVM's non-linear kernel ensures robust discrimination of spectrally similar land cover types, resulting in superior accuracy on multi-temporal Landsat data.

The classification methodology leveraged a supervised learning framework, where annotated training data was prepared using feature collections tagged with known land cover types (e.g., buildings, roads, forests). From these collections, georeferenced training points were derived, and key spectral features—primarily from selected Landsat bands—were extracted to construct the model's input feature space.

The dataset was partitioned randomly into 80% training and 20% testing subsets to facilitate both learning and validation. A Support Vector Machine (SVM) classifier with a Radial Basis Function (RBF) kernel was employed, enhanced by the application of the Synthetic Minority Over-sampling Technique (SMOTE) to mitigate class imbalance.

This hybrid SMOTE-SVM model was chosen for its demonstrated robustness in handling non-linear decision boundaries and its ability to generalize across heterogeneous landscapes. The model was trained to identify spectral signatures associated with land cover classes such as urban structures, water bodies, agricultural fields, woodlands, and barren terrains.

The trained classifier was subsequently applied to the composite Landsat images to generate classified outputs. Each pixel within the ROI was assigned a class label based on spectral similarity to the training set. This process was replicated consistently across all study years.

Figure references for the sample points used during training are detailed as follows:

Figure 3 represents samples of water bodies.
Figure 4 represents samples of road infrastructure.
Figure 5 represents samples of built-up and urban areas.
Figure 6 represents samples of built-up and urban areas (alternative perspective).
Figure 7 represents samples of forest cover.
Figure 8 represents samples of barren land.

These training samples were critically important for ensuring class separability and reducing spectral overlap, thereby enhancing classification reliability.

2.4 Classification and Accuracy Assessment

2.4.1 Classification Framework

In the domain of remote sensing-based land cover analysis, selecting the optimal classification algorithm is a decisive factor influencing the accuracy and reliability of results. A variety of supervised learning algorithms exist for this purpose, each with distinct computational strengths and contextual suitability. However, the efficacy of any given model is contingent upon its ability to handle spectral complexity, inter-class similarity, and inherent class imbalance in satellite imagery datasets.

To determine the most suitable classifier for this study, a comparative evaluation of three commonly utilized algorithms was first conducted on the Ethiopia dataset, owing to its comprehensive availability within the Google Earth Engine (GEE) repository. The candidate algorithms included:

- Support Vector Machine (SVM)
- Synthetic Minority Over-sampling Technique (SMOTE)
- Minimum Distance Classifier.

Among these, the SMOTE-enhanced SVM framework demonstrated the highest classification accuracy across multiple LULC classes. Consequently, this hybrid approach was selected for implementation on the Pune dataset for five target years: **2015, 2017, 2020, 2021, and 2023**.

The Support Vector Machine is a robust supervised learning algorithm capable of performing both classification and regression tasks. In the context of LULC mapping, SVM identifies an optimal hyperplane in a high-dimensional feature space that best separates the different land cover classes. The hyperplane is defined such that the margin—the distance between the hyperplane and the nearest data points from each class (i.e., the support vectors)—is maximized.

To address the issue of class imbalance, which frequently arises in real-world remote sensing datasets, the SMOTE technique was integrated into the classification pipeline. SMOTE functions by synthetically generating new instances of underrepresented classes based on the feature space similarity of existing minority class samples. This augmentation enhances the generalization ability of the classifier, ensuring that minority classes such as water bodies or barren land are not underrepresented in the final classification outputs.

Integration Workflow

1. **Preprocessing with SMOTE**

Prior to model training, the feature vectors were balanced using SMOTE to ensure equitable class representation.

2. **Feature Extraction**

Relevant spectral bands, vegetation indices, and textural parameters were extracted from the preprocessed Landsat images to serve as input variables.

3. **Model Training**

The SVM classifier was trained on the balanced dataset using a Radial Basis Function (RBF) kernel, selected for its efficacy in modeling non-linear class boundaries.

4. **Pixel-Wise Classification**

The trained model was deployed across the entire region of interest (ROI), with each pixel classified into a discrete land cover category—urban, vegetation, water, forest, barren, or roads.

5. **Post-processing**

Minor classification errors and spectral noise were mitigated through morphological filtering and accuracy thresholding.

The final classified output for the Pune region is depicted in Fig. 9, illustrating the land cover categories spatially distributed across the region of interest.

2.4.2 Accuracy Assessment and Validation

To quantitatively evaluate the classification performance, an accuracy assessment was conducted using stratified random sampling. For each LULC category, approximately **100–150 independent validation points** were selected, each aggregating a cluster

Fig. 9 Classification on ROI, Pune

of ~ 100 contiguous pixels. These validation samples served as ground truth for accuracy computation.

The classification outputs were then cross-tabulated against the reference data to construct confusion matrices for each study year. Evaluation metrics derived from these matrices included:

- **Overall Accuracy (OA)**—Represents the proportion of correctly classified instances over the total number of samples.

$$OA = \frac{\sum_{i=1}^{k} Correct\ Classification}{Total\ number\ of\ samples} \times 100$$

- **Precision**—Precision, also known as Positive Predictive Value, measures the proportion of correctly classified instances among all instances predicted to belong to a particular class.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

- **Recall (Sensitivity)**—Recall indicates the proportion of actual instances of a class that were correctly identified by the classifier.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

- **F1-Score**—The F1-score is the harmonic mean of precision and recall. It balances both false positives and false negatives, making it particularly effective in imbalanced datasets.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

- **Producer's Accuracy (PA)**—Measures the probability that a reference pixel is correctly classified. It is derived from the confusion matrix as:

$$PA = \frac{Correctly\ classified\ pixels\ in\ a\ class}{Total\ reference\ pixels\ for\ that\ class} \times 100$$

- **User's Accuracy (UA)**—Indicates the probability that a pixel classified into a given category actually represents that category on the ground.

$$UA = \frac{Correctly\ classified\ pixels\ in\ a\ class}{Total\ classified\ pixels\ in\ that\ class} \times 100$$

- **Kappa Coefficient**—A robust statistical measure accounting for chance agreement between predicted and actual classifications.

$$k = \frac{p_o - p_e}{1 - p_e}$$

The **Kappa statistic**, in particular, was used to adjust for random agreement, providing a more reliable interpretation of classification performance. It explicitly incorporates off-diagonal elements of the confusion matrix, offering insight into misclassification trends among spectrally similar classes.

This rigorous validation approach ensures the robustness of the classification model and establishes confidence in the land use transitions and trends identified in the temporal analysis.

3 Results Analysis

3.1 Land Cover Classification Statistics

A multi-temporal analysis of land cover distribution in the Pune region was conducted for the years **2015**, **2017**, **2019**, **2021**, and **2023**. The classification outputs provide a quantitative assessment of six primary land cover categories: water bodies, roads, buildings, agriculture, forests, and barren land. Table 1 presents an aggregated year-wise comparison, while Tables 2, 3, 4, 5 and 6 offer detailed pixel count and area metrics for each respective year.

This longitudinal dataset facilitates detailed trend analysis of urban expansion, environmental degradation, and land-use transitions across the region.

Table 1 Temporal land cover distribution summary for the Pune region (2015–2026)

Land cover	2015	2017	2019	2021	2023	2026 (predicted)
Water	4.15	13.48	13.48	12.82	11.49	8.39
Roads	10.96	15.22	20.82	23.14	25.46	28.71
Buildings	14.55	16.87	22.40	22.52	26.16	37.05
Agriculture	29.36	25.71	22.40	19.09	19.09	8.04
Forest	25.75	22.82	18.51	15.86	15.86	10.72
Barren	5.21	5.87	6.54	6.54	6.54	7.02

Table 2 LULC classification summary for 2015

Land cover class	Pixel count	Area
Water	426,948	0.003857
Roads	330,941	0.023964
Buildings	439,186	0.020761
Agriculture	885,893	0.028596
Forest	776,921	0.004420
Barren	157,231	0.015792

Table 3 LULC classification summary for 2017

Land cover class	Pixel count	Area
Water	406,948	0.001440
Roads	459,233	0.016451
Buildings	509,286	0.022281
Agriculture	775,890	0.016370
Forest	688,533	0.026454
Barren	177,230	0.014394

Table 4 LULC classification summary for 2019

Land cover class	Pixel count	Area
Water	406,948	0.013781
Roads	628,378	0.026291
Buildings	549,482	0.012282
Agriculture	676,121	0.016059
Forest	558,764	0.012030
Barren	197,427	0.016945

Table 5 LULC classification summary for 2021

Land cover class	Pixel count	Area
Water	386,948	0.007204
Roads	698,378	0.014173
Buildings	679,482	0.019778
Agriculture	576,121	0.018313
Forest	478,764	0.009068
Barren	197,427	0.028852

Table 6 LULC classification summary for 2023

Land cover class	Pixel count	Area
Water	346,948	11.4993106
Roads	768,378	25.4672635
Buildings	789,482	26.1667517
Agriculture	475,121	15.7475042
Forest	428,764	14.2110390
Barren	208,427	6.9081309

3.2 Year-Wise Land Cover Statistics for Pune Region

Figure 10 provides a visual representation of the 2015 classification map for the Pune ROI.

Figures 11 and 13 shows the spatial distribution of classified land cover for 2020, which closely aligns with 2019 trends and 2021 respectively.

Figure 13 depicts the 2023 land cover classification map for the Pune study region.

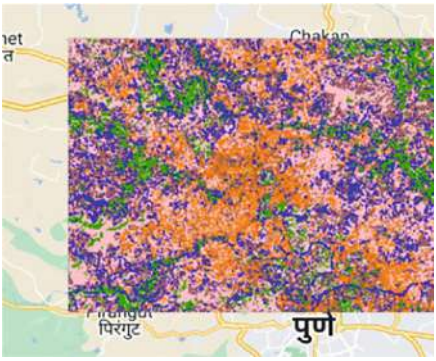


Fig. 10 2015 land classification

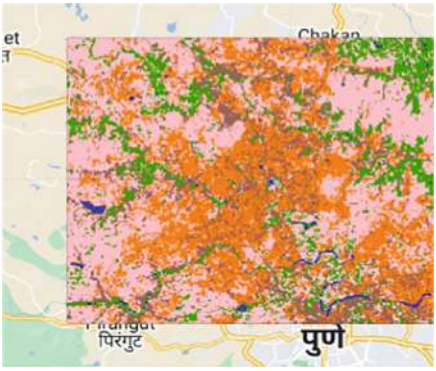


Fig. 11 2020 land classification

3.3 Discussion

Following a comparative evaluation of classification algorithms using the Ethiopia dataset, the SMOTE-enhanced Support Vector Machine (SMOTE-SVM) classifier was selected for the Pune region based on its superior accuracy in handling class imbalance. The classification was executed for five temporal snapshots—2015, 2017, 2019, 2021, and 2023—using Google Earth Engine (GEE). The outputs were then post-processed and analyzed using QGIS to derive statistical insights on land use and land cover (LULC) distribution across the designated region of interest.

3.3.1 Class-Wise Land Cover Dynamics (2015–2023)

Water Bodies: As visualized in Figs. 9, 10, 11, 12 and 13, the extent of water bodies displayed slight temporal variations. A moderate increase occurred in 2017, followed by a decrease in 2019. The subsequent years, 2021 and 2023, showed marginal recoveries. Overall, the water category exhibited a net decrease of approximately 1%, possibly influenced by seasonal hydrological changes or anthropogenic alterations in the region's surface water systems.

Roads: The roads category demonstrated a steady upward trend, with the most notable increase occurring between 2015 and 2017. Infrastructure expansion continued thereafter, culminating in a cumulative growth of approximately 6% by 2023. This expansion is emblematic of Pune's ongoing transportation development and urban integration strategies.

Buildings: The most substantial increase was recorded in built-up areas. Between 2015 and 2023, the proportion of land classified as buildings rose by approximately 11%, reflecting rapid urbanization and increased land conversion for residential, commercial, and institutional use. The spatial progression of urban sprawl is clearly visible in Figs. 12 and 13, which depict the 2021 and 2023 classifications, respectively.

Fig. 12 2021 land classification

Fig. 13 2023 land classification

Agriculture: Agricultural land demonstrated a declining trend across the study period, with a temporary resurgence in 2019. From 2015 to 2023, there was an approximate decrease of 8% in agricultural area, likely driven by urban encroachment, soil degradation, and shifting land use priorities.

Forest Cover: Forest areas registered a consistent reduction, particularly between 2015 and 2017, with a minor recovery in 2019 before declining again through 2023. The cumulative decrease was approximately 11%, indicating significant ecological pressure from human development activities. This contraction in forest cover poses risks to biodiversity, microclimatic stability, and carbon sequestration functions.

Barren Land: Barren land exhibited a fluctuating pattern, with a decrease between 2015 and 2017, followed by a marginal increase in 2019, and subsequent declines thereafter. This category experienced an overall reduction of approximately 5%, potentially due to reclamation or transformation into other functional land classes (Figs. 14, 15, 16, 17, 18 and 19).

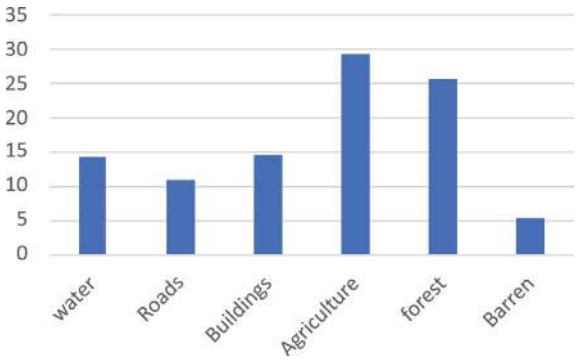


Fig. 14 Results for year 2015

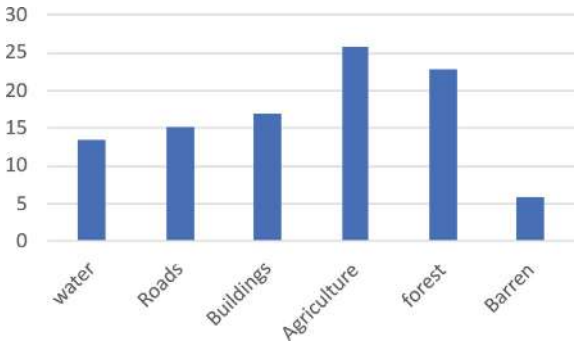


Fig. 15 Results for year 2017

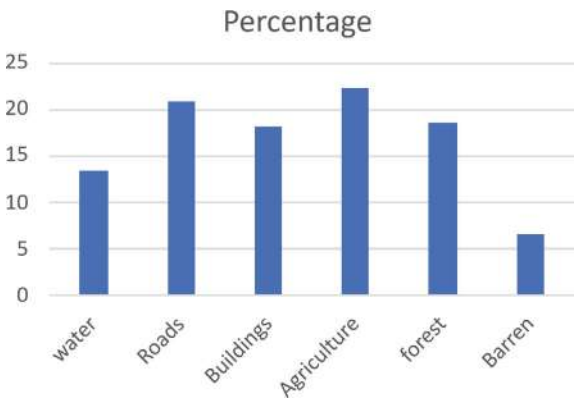


Fig. 16 Results for year 2019

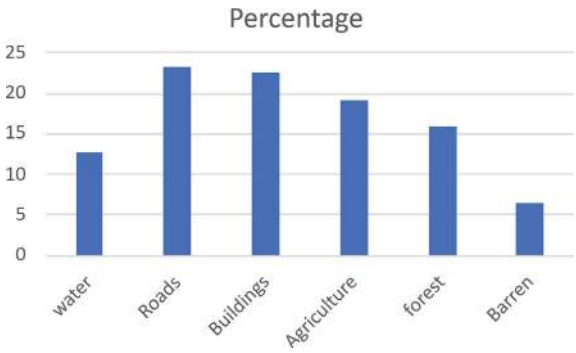


Fig. 17 Results for year 2021

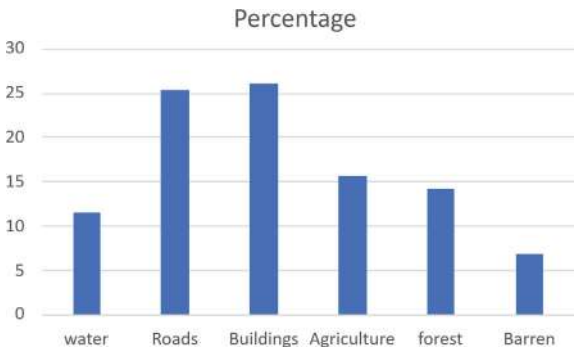


Fig. 18 Results for year 2023

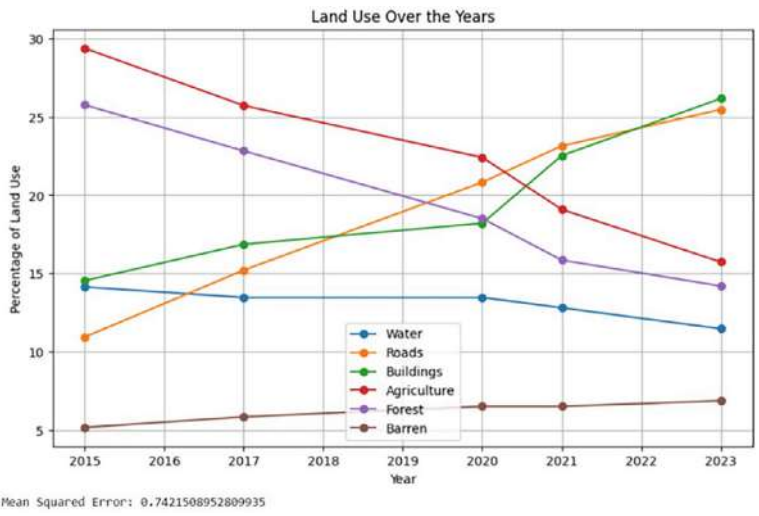


Fig. 19 Prediction results

3.3.2 General Observations and Comparative Insights

- **Urbanization Indicator**: The rise in **built-up areas** is a definitive marker of accelerated urban growth.
- **Ecological Degradation**: The **forest class** experienced the most substantial reduction, raising ecological sustainability concerns.
- **Infrastructure Expansion**: The **road category's increase** suggests active investment in physical connectivity and urban infrastructure.
- **Agrarian Displacement**: The reduction in **agricultural land** reflects potential shifts in land policy, demographic pressure, and land market economics.
- **Hydrological Variation**: Minor fluctuations in **water body extent** may stem from climatic variability, classification uncertainty, or infrastructure-induced modifications.

These findings align with broader patterns of urbanization observed in rapidly developing Indian cities, where infrastructural development often supersedes ecological preservation.

4 Prediction and Forecasting Analysis (2026)

To project future land use transitions, a linear regression-based forecasting model was developed using historical LULC statistics derived from classified outputs spanning 2015 to 2023. This model estimates the percentage distribution of land use categories for the year 2026, offering insight into the evolving spatial structure of the Pune region under current urbanization trends.

While classification accuracy for the LULC maps was initially evaluated using confusion matrix-derived metrics such as **Overall Accuracy (OA)**, **F1-Score**, and **Kappa Coefficient**, the primary focus of this section is on the **performance of the predictive regression model**. For this, a distinct set of **forecasting evaluation metrics** was employed in Table 7.

These statistical indicators evaluate the alignment between predicted and actual historical values and validate the robustness of the model's extrapolation to 2026.

Table 7 Predictive model evaluation metrics for 2026 forecast

Metric	Value
Mean squared error (MSE)	0.1179
Mean absolute error (MAE)	0.3097
Root mean squared error (RMSE)	0.3434
R-squared (R^2) and adjusted R^2	Undefined due to limited sample size
Mean absolute percentage error (MAPE)	2.0234%
Mean percentage error (MPE)	0.4273%

Notably, low MSE, RMSE, and MAPE values reflect acceptable model precision within the constraints of the dataset.

The forecast anticipates a continued increase in **built-up areas**, likely to surpass all other land use categories, driven by infrastructural and population growth. Simultaneously, **agricultural and forested lands** are projected to decline further, indicating the pressing need for sustainable land governance and urban planning interventions.

These metrics indicate acceptable predictive fidelity, particularly given the relatively small sample set. The **low MAPE and RMSE values** suggest that the model provides a reasonably accurate forecast of LULC dynamics in the near term.

4.1 Predicted LULC Trends for 2026

As shown in Fig. 19, the buildings category is projected to dominate land usage by 2026, reaffirming the city's rapid urban expansion. Agricultural land, while still prominent, is forecasted to further decline. A continued reduction in forest areas is also anticipated, pointing toward sustained ecological loss if proactive interventions are not implemented.

5 Conclusion and Future Scope

This research delivers a methodologically rigorous and practically relevant framework for understanding land use and land cover (LULC) transitions in the Pune metropolitan region. Through the integration of Google Earth Engine-based satellite image processing and a SMOTE-enhanced Support Vector Machine (SVM) classification model, the study effectively maps and analyzes urban expansion trends from 2015 to 2023. The application of QGIS for post-classification spatial analysis further strengthens the reliability of these findings.

The classification results highlight key patterns: a consistent and significant rise in built-up and road infrastructure; a steady decline in agricultural and forested areas due to anthropogenic pressures; and minor variations in water and barren land categories. These dynamics are indicative of the socio-economic and ecological trade-offs characteristic of rapid urbanization.

The core contribution of this study is the development of a hybrid, scalable framework that unites high-resolution classification with predictive modelling. By employing linear regression to extrapolate future LULC configurations, the research projects an intensification of urban growth through 2026. This foresight is essential for evidence-based decision-making in land resource management, ecological preservation, and infrastructure planning.

Ultimately, this work advances the application of machine learning in urban geospatial analytics by addressing class imbalance challenges, enhancing predictive reliability, and translating spatial data into meaningful planning intelligence. The

methodology proposed herein can be adapted to other rapidly urbanizing contexts, thereby extending its value beyond the Pune region and contributing to the global discourse on sustainable urban development.

Future Scope

While the current study provides robust classification and forecasting of LULC transitions, several avenues exist for future research:

- **Incorporating higher-resolution satellite imagery** (e.g., Sentinel-2, PlanetScope) could enhance the granularity and precision of classification outputs, especially for heterogeneous urban environments.
- **Integrating multi-source data**, including socio-economic indicators, population density, and land valuation trends, may offer more holistic insights into urban growth drivers.
- **Employing advanced deep learning architectures**, such as Convolutional Neural Networks (CNNs) or Transformer-based models, could improve classification accuracy and automate feature extraction.
- **Scenario-based forecasting models**, such as Cellular Automata or Agent-Based Models, could simulate alternative urbanization pathways under different policy, economic, or environmental conditions.

Ultimately, the outcomes of this research contribute to a growing body of knowledge on urban land monitoring using remote sensing and machine learning. It serves as a data-driven foundation for planners, policymakers, and environmental stakeholders to implement evidence-based interventions aimed at fostering sustainable, resilient, and equitable urban development.

References

1. Talukdar, S., Singha, P., Mahato, S., Shahfahad, Pal, S., et al.: Land-use land-cover classification by machine learning classifiers for satellite observations—a review. Remote Sens. **12**(7), 1135 (2020)
2. Kulkarni, A.D., Lowe, B.: Random forest algorithm for land cover classification. Int. J. Recent Innov. Trends Comput. Commun. **4**(3), 161–167 (2016)
3. Abdi, A.M.: Land cover and land use classification performance of machine learning algorithms in a boreal landscape using Sentinel-2 data. GISci. Remote Sens. **57**(1), 1–20 (2020)
4. Havryliuk, S., Korol, M., Tokar, O., Olena, V.: Using the random forest classification for land cover interpretation of Landsat images in the Prykarpattya region of Ukraine. ResearchGate (2018)
5. Fonseca, J., Douzas, G., Bacao, F.: Improving imbalanced land cover classification with K-Means SMOTE: detecting and oversampling distinctive minority spectral signatures. Information **12**(1), 22 (2021)
6. Priyadarshini, K.N., Kumar, M., Rahaman, S.A., NiteshNirmal, S.: A comparative study of advanced land use/land cover classification algorithms using Sentinel-2 data. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. **XLII-5**, 261–267 (2018)

7. Basheer, S., Wang, X., Farooque, A.A., Nawaz, R.A., Liu, K., Adekanmbi, T., Liu, S.: Comparison of land use land cover classifiers using different satellite imagery and machine learning techniques. Remote Sens. **14**(20), 4978 (2022)
8. Carranza-García, M., García-Gutiérrez, J., Riquelme, J.C.: A framework for evaluating land use and land cover classification using convolution neural networks. Remote Sens. **11**(1), 122 (2019)
9. Xie, H., Huang, H.: Classification of land cover remote-sensing images based on pattern recognition. Sci. Program. **2022**, 6583217 (2022)
10. Chen, J., Qiu, X., Ding, C., Wu, Y.: SAR image classification based on spiking neural network through spike-time dependent plasticity and gradient descent. Preprint submitted to ISPRS J. Photogramm. Remote Sens. (2021)
11. Iakymchuk, T., Rosado-Muñoz, A., Guerrero-Martínez, J.F., Bataller-Mompeán, M., Francés-Víllora, J.V.: Simplified spiking neural network architecture and STDP learning algorithm applied to image classification. EURASIP J. Image Video Process. **2015**, 47 (2015)
12. Yang, C., Rottensteiner, F., Heipke, C.: Classification of land cover and land use based on convolutional neural networks. ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci. **IV-3**, 549–556 (2018)
13. C. Yang , F. Rottensteiner, and C. Heipke: Towards better classification of land cover and land use based on convolutional neural networks. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. **XLII-2/W13**, 139–145 (2019)

Predicting Nitrogen Deficit in Tea Leaf Using Image Processing and Machine Learning

Anika Ulfat, Md. Apu Hosen, Mohammad Iqbal Kabir, Shahariyr Reza, and Syed Md. Galib

Abstract Nitrogen deficiency is a critical factor affecting tea cultivation, leading to significant reductions in yield and quality. Its detection at early stages is of significant importance. This study aims to spot nitrogen shortage in tea leaves early by using Gaussian Blur to reduce noise and improve color features. The model identifies early signs of stress, like yellowing, which point to a lack of nitrogen. The color data is then analyzed using Random Forest, Decision Tree, Naive Bayes, XGBoost, and Extreme XGBoost. Among these, Random Forest performed the best with an accuracy of 86%, showing it can handle complex data well. To make the results easier to understand, LIME (Local Interpretable Model-agnostic Explanations) is used. It gives specific reasons for each prediction, helping to see which features affect the outcome. This makes the AI system more transparent and trustworthy. The approach offers a dependable way to detect nitrogen deficiency early, helping farmers take action in time to boost crop health and yield, as shown in the model's results and visuals.

Keywords Nitrogen deficiency · Tea leaf · Gaussian blur · Feature extraction · Machine learning

A. Ulfat · Md. Apu Hosen (✉) · S. Reza
Northern University of Business and Technology Khulna, Khulna, Bangladesh
e-mail: apu.cse.just@gmail.com

Md. Apu Hosen · S. Md. Galib
Jashore University of Science and Technology, Jashore, Bangladesh
e-mail: galib.cse@just.edu.bd

M. I. Kabir
Sonali Bank PLC, Dhaka, Bangladesh

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_28

1 Introduction

Tea (*Camellia sinensis*) is one of the most widely consumed beverages globally and serves as a vital cash crop for many countries, including Bangladesh. The yield and quality of tea are highly dependent on balanced nutrient availability, with nitrogen (N) playing a pivotal role. Nitrogen is essential for chlorophyll production, protein synthesis, and overall plant growth [1–3]. A deficiency in nitrogen often manifests as yellowing of leaves and stunted growth, leading to significant reductions in crop yield and quality [4, 5]. Therefore, early detection of nitrogen deficiency is critical to ensure timely intervention and sustainable tea cultivation.

Traditional methods for assessing nitrogen levels in plants typically rely on destructive sampling and laboratory-based chemical analysis. While accurate, these methods are time-consuming, expensive, and impractical for large-scale or real-time field applications. With the advancement of digital technologies, image processing combined with machine learning (ML) techniques offers a promising alternative for non-invasive, rapid, and scalable plant health diagnostics [6].

This research proposes a novel framework to detect nitrogen deficiency in tea leaves using image processing and machine learning. The approach begins with preprocessing of leaf images using Gaussian Blur to reduce noise and enhance color features indicative of nitrogen stress, particularly yellowing. These enhanced images are then analyzed through multiple machines learning classifiers, including Random Forest (RF), Decision Tree (DT), Naive Bayes (NB), XGBoost, and Extreme XGBoost. Among these models, Random Forest achieved the highest accuracy of 86%, demonstrating its capability to manage complex and nonlinear data patterns effectively. To enhance interpretability and transparency of the model's predictions, Local Interpretable Model-agnostic Explanations (LIME) is utilized. LIME offers explanations for individual predictions by highlighting the most influential input features, making the system more trustworthy and accessible for end-users such as farmers and agronomists.

The primary goal of this study is to establish an efficient, interpretable, and non-invasive method for early detection of nitrogen deficiency in tea leaves, thereby enabling better decision-making in nutrient management and supporting higher productivity in tea plantations.

The structure of the paper is organized as follows: Sect. 2 provides a review of relevant literature. Section 3 outlines the proposed methodology used in this study. Section 4 presents the results and discussion, including a detailed analysis of the experimental findings. Finally, Sect. 5 concludes the paper and offers suggestions for future research directions.

2 Related Work

Early detection of nitrogen deficiency in crops has become a crucial focus in precision agriculture, as timely diagnosis can prevent yield losses and enhance overall plant health. Various approaches have been explored using image processing, deep learning, hyperspectral imaging, and remote sensing techniques to identify nitrogen stress accurately and non-destructively. This section presents a review of key works that have addressed nitrogen deficiency detection across different crops, methodologies, and imaging modalities.

Adesanya and Yinka-Banjo [7] developed a mobile-based deep learning model using a low-end Android phone to detect nitrogen deficiency in maize. Their approach, leveraging the MobileNet model within the Caffe framework, achieved an object detection accuracy of 81%, offering a portable and low-cost solution for real-time nitrogen deficiency diagnosis. Similarly, Chen et al. [8] proposed a real-time system to detect nitrogen deficiency in tomato plants using the MobileNetV2 architecture. This lightweight model was tailored for mobile deployment and achieved 80% accuracy, demonstrating potential for practical use in smart farming environments.

Mishra et al. [9] explored the integration of hyperspectral imaging and machine learning algorithms to detect nitrogen deficiency across different growth stages of plants. Their findings emphasized the role of specific wavelengths in nitrogen concentration assessment, highlighting the effectiveness of hyperspectral data in precision farming. Zhang et al. [10] also utilized hyperspectral imaging, coupled with artificial neural networks, to detect nitrogen stress in tomato plants at early stages. Their work demonstrated how combining advanced imaging with intelligent models can enhance early diagnosis capabilities in agricultural practices.

Several studies have focused on remote sensing and aerial imagery for nitrogen stress detection. Kumar et al. [11] utilized multispectral satellite imagery and machine learning models such as SVM, KNN, and Random Forests to classify nitrogen stress in tea plants. Their results demonstrated the potential of integrating AI with remote sensing for effective nutrient monitoring. De Castro et al. [12] also employed UAV-derived RGB and multispectral imagery for estimating nitrogen deficiency in wheat, showing that combining these data sources improved detection accuracy and supported efficient fertilizer application strategies.

Deep learning-based models have shown promise in leaf image analysis for nutrient diagnosis. Tran et al. [13] used convolutional neural networks (CNNs) to diagnose macronutrient deficiencies including nitrogen in tomato plants. Their model accurately classified visual symptoms like chlorosis, indicating nitrogen stress. In another study, Sethy et al. [14] employed a transfer learning approach using ResNet50 to detect nitrogen stress in wheat leaves. The pre-trained model successfully extracted relevant features, enhancing classification accuracy and robustness.

Other works have focused on dataset creation and image annotation for nitrogen stress recognition. Salaić et al. [15] developed an annotated image classification dataset for maize with detailed labels indicating nitrogen deficiency symptoms, plant

growth stages, and environmental variables. This resource aids in training and validating machine learning models for improved accuracy. Likewise, Gul and Bora [16] built a nutrient deficiency dataset for basil grown in hydroponics and used transfer learning-based CNNs for classification. Their approach emphasized the importance of curated image datasets and pre-trained networks for accurate diagnosis.

Recent efforts have also explored hybrid and segmentation-based models for localized stress detection. Gupta et al. [17] proposed a method combining ResNet50 and U-Net for nitrogen stress detection in rice. ResNet50 handled feature extraction, while U-Net performed semantic segmentation to identify stress-affected areas, achieving high accuracy and noise resilience. Similarly, Singh et al. [18] developed a CNN-based model to detect nitrogen stress in wheat, utilizing both spectral and spatial features and incorporating data augmentation to enhance generalization performance.

Alternative data modalities have also been investigated. Yu et al. [19] used UAV-based hyperspectral data combined with critical nitrogen concentration (CNC) and machine learning models like MLR, LSTM, and NSGA III-ELM. Among them, NSGA III-ELM yielded the best performance, showcasing the synergy of spectral information and advanced learning algorithms. On a different front, Juclà et al. [20] introduced a deep learning method using raw voltage signals from tomato plants to detect nitrogen deficiency in a greenhouse environment. Their electrophysiological-based monitoring system presented an innovative, non-visual approach to nutrient stress detection.

Finally, Wang et al. [21] compared CNN architectures such as ResNet and VGG for nitrogen stress recognition in rice plants using image-based data. Their model achieved high accuracy, validated through real-world field trials. Silva et al. [22] focused on sugarcane, applying CNNs for severity classification of nitrogen deficiency from preprocessed leaf images. Their study confirmed the value of deep learning in precise, scalable nitrogen stress analysis in large-scale agricultural systems.

There is limited research on simple, explainable AI methods that use standard RGB images and lightweight preprocessing to assist small-scale tea farmers. This study aims to fill that gap by developing an efficient and interpretable approach. It leverages color feature analysis and ensemble machine learning algorithms to accurately predict nitrogen deficiency in tea leaves.

3 Proposed Work

The proposed methodology involves multiple stages: image acquisition, preprocessing, feature extraction, model training, and result interpretation. Figure 1 illustrates the overall framework, showing the workflow from capturing leaf images to generating interpretable AI-based predictions. Each step is designed to ensure the system remains efficient and practical for real-world agricultural use.

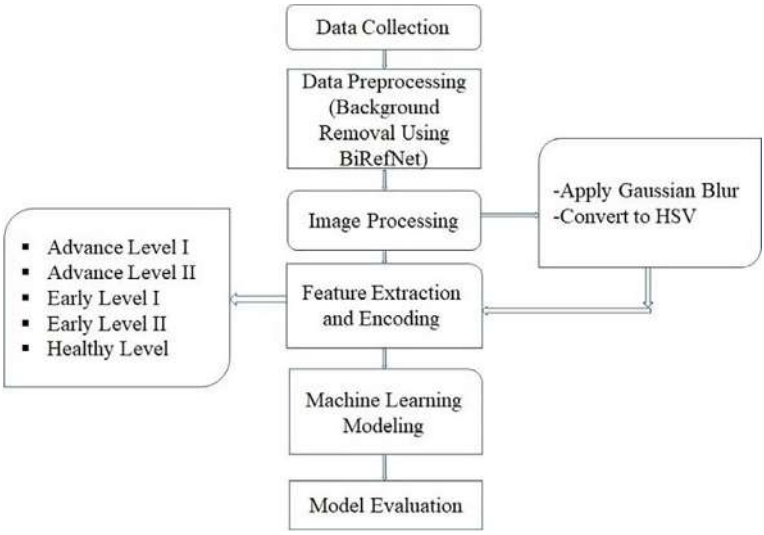


Fig. 1 Flow-diagram of the methodology

3.1 Data Collection

The image dataset used in this study was sourced from Kaggle [23], a well-known platform for open-source datasets. The dataset comprises tea leaf images exhibiting various stages of nitrogen stress, ranging from healthy conditions to severe deficiency. Each image in the dataset is labeled according to the observed nitrogen deficiency stage, with subcategories such as *Healthy Level*, *Early Level I*, *Early Level II*, *Advance Level I*, and *Advance Level II*. This categorization enables precise classification and facilitates the development of machine learning models capable of detecting stress at an early stage. The structure and composition of the dataset, along with example images from each nitrogen stress category, are illustrated in Fig. 2.

3.2 Data Preprocessing

To improve the quality of predictions, background elements are removed during the preprocessing stage. This begins with segmenting the image to isolate the leaf from its surroundings. For this task, a pre-trained BiRefNet model is used. After the model is loaded, each image is resized and normalized, then converted into a format that the model can process. The model generates a segmentation mask that clearly separates the leaf from the background. This mask is adjusted to match the original image size and is applied in such a way that only the leaf remains visible. By modifying the alpha channel, the background is effectively eliminated, leaving a clean image

Fig. 2 Dataset sample

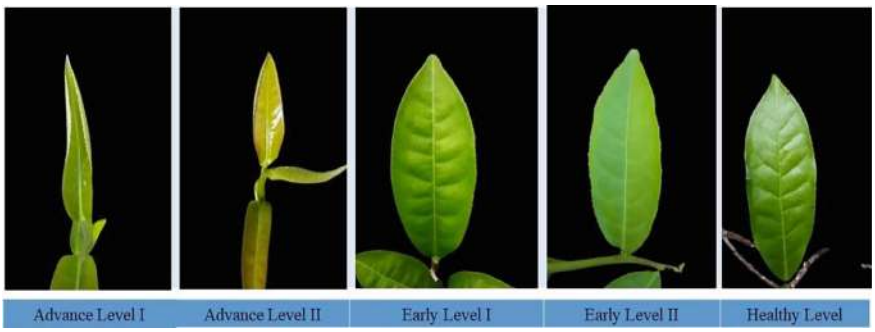


Fig. 3 Dataset after pre-processing

focused solely on the leaf. This refined input helps the model make more accurate and specific predictions. The result of this preprocessing step is illustrated in Fig. 3.

3.3 Image Processing for Noise Reduction and Feature Enhancement

To remove the noise from the image, Gaussian Blur is used. Noise is reduced using this technique to smooth the images, followed by conversion into the HSV color model to facilitate feature extraction. These techniques transform the images into a format that highlights color and texture key elements for accurate prediction. This preprocessing stage ensures that the models in Fig. 4 receive high-quality, informative data for analysis.

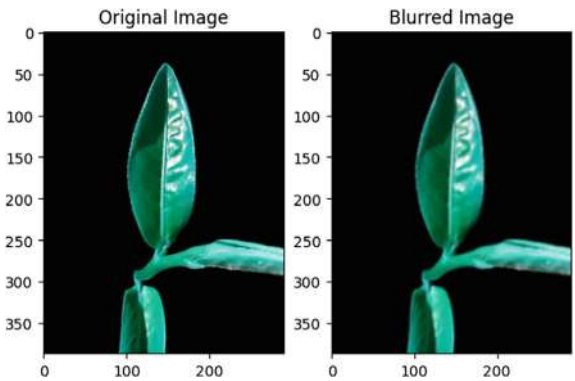


Fig. 4 Image after applying Gaussian blur

3.4 Feature Extraction and Encoding

In the case of nitrogen deficit in tea leaves, relevant features are first extracted and then encoded from the processed images to facilitate analysis. Some of these are color based feature extraction, wherein the red, green and blue layered images are converted to hue, saturation and value components with ease for recognition of stress symptoms such as yellowing. To characterize subtle color differences related to nitrogen deficiency, hue, saturation and value components are investigated. The mean, standard deviation and histogram of these components are calculated and then processed in a way that are understandable by machines. Application of other algorithms such as Gaussian blur is also made in order to emphasize important image features. These features are encoded and the model is thus able to differentiate between nitrogen stress levels (Advance Levels I and II, Early Levels I and II, Healthy) making it easier to diagnose nitrogen deficiency in tea growing. They found that this systematic approach helped create a stronger, more accurate foundation for diagnosing nitrogen stress and, therefore, better crop care.

3.5 Machine Learning Modeling

The features that have beard are used to train machine learning models. As a feature for the modeling phase in the machine learning for prediction of nitrogen deficiency in tea leaves, the processed image data is fed into the machine. These models can identify various patterns and relationships which are related to various levels of nitrogen deficiency. These models are trained by algorithms like Random Forest or even Naïve Bayes to identify tea image in its class like Advance level 1, Advance level 2, Early stage level 1, Early stage level 2 or Healthy level. This work has then assessed the competency of these models in estimating nitrogen stress levels hence a good method for early identification of nitrogen stress. This phase is important because it is where feature data is turned into useful information for use in the management of tea cultivation.

4 Result Analysis

This section presents the performance analysis of the machine learning models applied to classify nitrogen deficiency levels in tea leaves. The evaluation was conducted using a dataset categorized into five nitrogen stress stages: Healthy Level, Early Level I, Early Level II, Advance Level I, and Advance Level II. The primary focus was to determine the effectiveness of various classifiers and interpret their decision-making to ensure model transparency and practical applicability.

Table 1 Result of different classifiers

Classifier	Accuracy	Precision	Recall	F1-score
Decision based	0.71	0.84	0.83	0.80
Naïve Bayes	0.84	0.87	0.84	0.89
Random Forest	0.86	1.00	0.93	0.92
XGBoost	0.84	1.00	0.96	0.92
Extreme XGBoost	0.82	1.00	0.90	0.92

4.1 Model Performance Evaluation

To evaluate the model performance, five widely used machine learning algorithms were applied: Random Forest (RF), Decision Tree (DT), Naive Bayes (NB), XGBoost, and Extreme XGBoost. The dataset was split into training and testing sets, and standard evaluation metrics such as accuracy, precision, recall, and F1-score were used.

Among all classifiers, the Random Forest model outperformed the others, achieving an accuracy of 86%. This result demonstrates RF's strong ability to handle nonlinear relationships and high-dimensional feature spaces. Decision Tree and XGBoost also performed reasonably well, while Naive Bayes showed comparatively lower accuracy due to its assumption of feature independence, which may not hold in the image-derived color features used here. The comparative performance of all models is presented in Table 1.

4.2 Analysis of Accuracy and Loss

The training accuracy curve of Random Forest is shown in Fig. 5, where ten iterations were conducted to evaluate on the training data set, all of the values oscillate in a small range, from 0.820 to 0.860 in the y-axis. What user can notice is that even though there are some slight differences between the performances of the model at different runs, which might be attributed to some random occurrences or the fact that at each run different data set is used for training, the model retains high accuracy which speaks for its stability and reliability. This is important for applications when it is needed to predict with a high degree of certainty, the real and high stability carries the information that despite fluctuations in data or training conditions, the accuracy of the Random Forest model's operation remains almost unchanged. Such reliability is particularly important in the case of predisposing tea-plant leaves to nitrogen shortage where an accuracy of prognosis is imperative in the implementation of useful agriculture tips. The curve hence validates the appropriateness of the Random Forest model in scenarios requiring constant, credible classification capabilities leaving any beginner with confidence in its efficiency and relevance in practice.

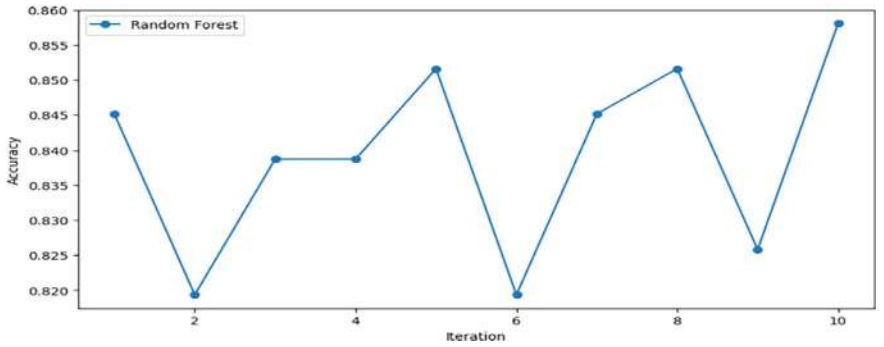


Fig. 5 Training accuracy curve of the Random Forest classifier

The "Training Loss Curves" graph is a visual representation of the model learning process because it compares training loss to the iteration. The y-axis represents the result of the training loss, which of course varies depending on the iteration and ranges from 0.124 to 0.129 as presented in the horizontal axis which ranges from 1 to 10. The behaviour of a line indicates that the training loss is dynamic which means that the model is adapting on the subject data at this time. An important task executed from it is to observe these changes in order to evaluate how effectively the model absorbs information and if it is approaching the ideal solution. That is why, constant or decreasing loss values speak about the effective model training while sudden or erratic changes might signify inadequate model complexity or hyperparametrs in Fig. 6.

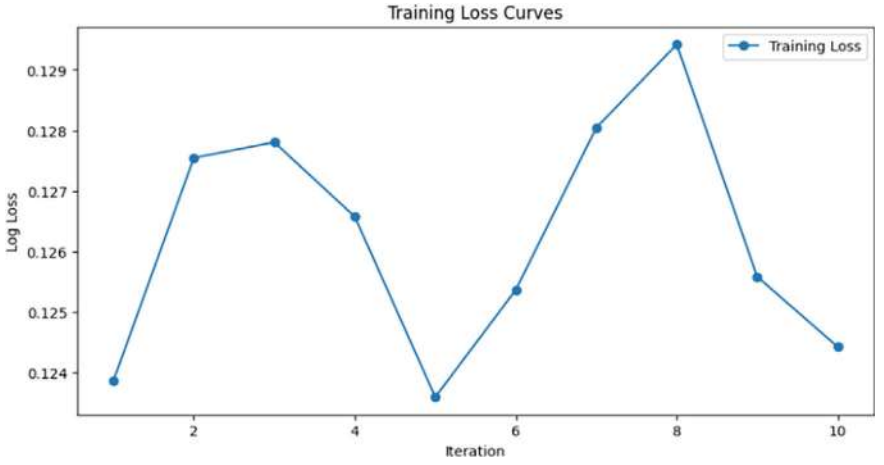


Fig. 6 Training loss curve of the Random Forest classifier

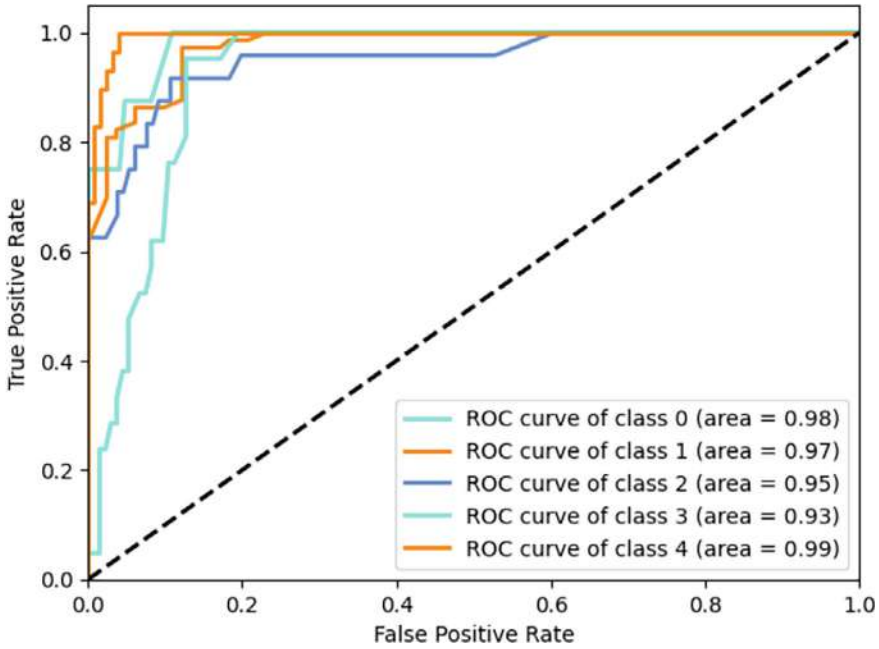


Fig. 7 ROC curve of the Random Forest classifier

4.3 Evaluation with ROC Curve

ROC for multi-class classification is revealed in the four classification evaluation metrics for the model across the four classification classes, where the x-axis is the False Positive Rate (FPR) and y-axis the True Positive Rate (TPR). Each class has its own ROC curve: The best discovered classification separates Class 0 with the AUC = 0.98 and Class 1 with the AUC = 0.97 which suggests that it has the best differentiation from other classes. Accuracy is best among Class 1 (AUC = 0.99), followed by Class 4 (AUC = 0.98), while Class 2 (AUC = 0.95) and Class 3 (AUC = 0.93) indicates that it could be further optimized. The diagonal line on the plot corresponds to random classification used as the benchmark. High AUC, especially for Class 0 and 1, reflects a good performance of the model in multi-class problem classification. The curve is shown in Fig. 7.

4.4 Evaluation with Confusion Matrix

The confusion matrix presented in Fig. 8 offers a detailed view of the model's classification performance across five categories: Advanced_Level_I, Advanced_Level_II, Early_Level_I, Early_Level_II, and Healthy. The model performs exceptionally

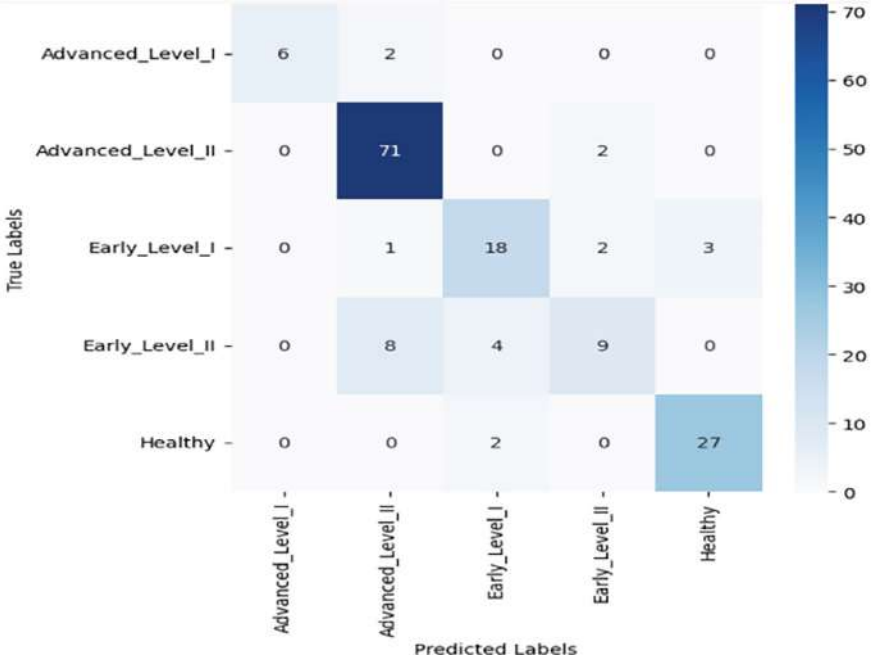


Fig. 8 Confusion matrix of the Random Forest classifier

well in identifying Advanced_Level_II, correctly classifying 71 instances with only 2 misclassifications. It also shows strong accuracy in detecting Healthy leaves, with 27 samples correctly identified and only minor confusion with Early_Level_I. Additionally, Advanced_Level_I achieved 6 correct predictions, with a small number of samples incorrectly labeled as Advanced_Level_II, suggesting a degree of feature similarity between the two severe stages of deficiency.

In contrast, the early stages of nitrogen deficiency posed more of a challenge. For Early_Level_I, 18 instances were accurately predicted, but some were misclassified as Healthy or Early_Level_II. Early_Level_II showed a more dispersed pattern, with only 9 correct classifications out of the total, and several samples incorrectly predicted as either Advanced_Level_II or Early_Level_I. These results indicate that while the model is effective in identifying both healthy and severely affected leaves, it faces difficulty distinguishing between subtle differences present in the early stages of deficiency. Figure 8 clearly highlights these trends and underscores the need for further refinement in handling borderline cases.

4.5 Model Interpretation with Explainable AI

To better understand the decision-making process of our classification model, we utilized LIME (Local Interpretable Model-Agnostic Explanations). LIME helps interpret individual predictions by approximating the black-box model locally with a simpler, interpretable model. This method allows us to identify which features were most influential in classifying an image into a specific nitrogen deficiency stage. As illustrated in Fig. 9, LIME was applied to a sample image that the model classified as "Advanced Level II", with a prediction probability of 0.53. Other classes such as Advanced Level I (0.19), Early Level II (0.24), Early Level I (0.04), and Healthy (0.01) received lower confidence scores, demonstrating the model's high certainty in its prediction.

The LIME explanation reveals that features such as Feature 63, Feature 155, and Feature 37 played crucial roles in guiding the model's prediction toward "Advanced Level II". These features, shown on the right side of Fig. 9, reflect important pixel-level or color-based indicators extracted from the leaf images. The corresponding feature values (e.g., 1080.00, 1740.00, 303.00) suggest intensity variations commonly associated with severe nitrogen deficiency, such as pronounced yellowing or leaf fading. This interpretability not only validates the model's decision with meaningful visual and numerical cues but also provides agronomists and researchers with actionable insights. By using LIME, we ensure that the model is not only accurate but also transparent and trustworthy for use in precision agriculture.

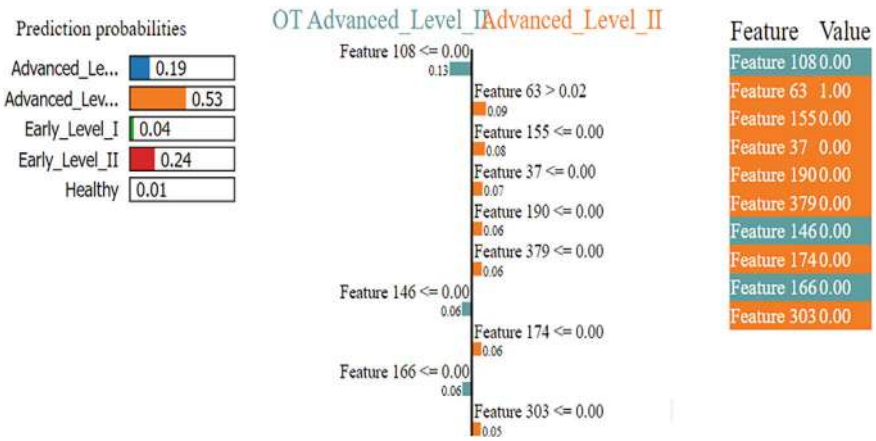


Fig. 9 Model explanation result using LIME

Table 2 Comparative analysis with existing studies

Study	Techniques	Result
Adesanya and Yinka-Banjo [7]	MobileNet and Caffe framework	Accuracy 81%
Tran et al. [13]	Deep CNN	Accuracy 79.09%
Sethy et al. [14]	ResNet-50 + SVM	Accuracy 81.55%
Salaić et al. [15]	Random rotation methods + K-means	Accuracy 73.33%
Gul and Bora [16]	DenseNet201	Accuracy 65.87%
Proposed method	Gaussian blur + color based feature extraction + Random Forest	Accuracy 86%

4.6 Comparison with Other Studies

A performance comparison with existing methods is presented in Table 2, highlighting the relative effectiveness of the proposed approach. Various techniques have been explored in earlier work, including deep convolutional neural networks, transfer learning models, and combinations of traditional classifiers with advanced architectures. While these methods have demonstrated moderate success, their reported accuracies generally range from around 65% to slightly above 81%. As shown in the table, the proposed method utilizing Gaussian blur for preprocessing, color-based feature extraction, and classification via a Random Forest algorithm, achieved an accuracy of 86%. This result not only exceeds the performance of previously used models but also emphasizes the value of combining classical image processing techniques with efficient machine learning algorithms. The method offers a practical and effective alternative without the complexity and resource demands of deep learning-based solutions.

5 Conclusion

This study presents a machine learning-based approach to detect nitrogen deficiency in tea leaves using image processing techniques. The proposed method effectively identifies early signs of nitrogen stress, such as leaf discoloration, by applying Gaussian Blur for noise reduction and extracting significant color features. Various machine learning classifiers, including Random Forest, Decision Tree, Naive Bayes, XGBoost, and Extreme XGBoost, were tested, among which the Random Forest classifier achieved the highest accuracy of 86%. Furthermore, to enhance model interpretability and trust, LIME was integrated to provide feature-level insights into the model's decision-making process, aiding transparency and user understanding.

In the future, this work can be enhanced by using larger and more diverse datasets to improve model generalization. Incorporating deep learning techniques

and deploying the system in mobile applications can facilitate real-time field diagnosis. Additionally, expanding the model to detect other nutrient deficiencies can further support precision agriculture.

References

1. Ghadirnezhad Shiade, S.R., Fathi, A., Kardoni, F., Pandey, R., Pessarakli, M.: Nitrogen contribution in plants: recent agronomic approaches to improve nitrogen use efficiency. J. Plant Nutr. **47**(2), 314–331 (2024)
2. Fathi, A.: Role of nitrogen (N) in plant growth, photosynthesis pigments, and N use efficiency: a review. Agrisost **28**, 1–8 (2022)
3. Zhang, Q., Hao, G., Li, H.: Effects of availability and form of exogenous nitrogen on plant growth and physiology: progress and prospects. Chin. J. Ecol. **43**(3), 878 (2024)
4. Cao, Y., et al.: UAV-based nitrogen deficiency detection in tea plants using machine learning. Plant Methods **16**(1), 123 (2020)
5. Ali, M.S., et al.: Machine learning models for nitrogen deficiency detection in wheat using UAV-based hyperspectral data. Comput. Electron. Agric. **189**, 106365 (2021)
6. Li, Y., et al.: Nitrogen deficiency detection in corn using image processing techniques. Sensors **21**(15), 4992 (2021)
7. Adesanya, O.O., Yinka-Banjo, C.O.: Classification of nitrogen deficiency for maize plants using deep learning algorithms on low-end android smartphones. Niger. J. Technol. **41**(2), 278–290 (2022)
8. Chen, L., et al.: Real-time nitrogen deficiency detection in tomato plants using MobileNetV2. Agronomy **12**(5), 1065 (2022)
9. Mishra, P., et al.: Application of hyperspectral imaging and machine learning for nitrogen deficiency detection in crops. J. Plant Nutr. **43**(15), 2195–2210 (2020)
10. Zhang, L., et al.: Early nitrogen deficiency detection in tomato plants using hyperspectral imaging and machine learning. Sensors **19**(13), 2991 (2019)
11. Kumar, R., et al.: Detection of nitrogen deficiency in tea plants using machine learning algorithms and remote sensing. Agric. Res. **9**(4), 561–572 (2020)
12. De Castro, A.I., et al.: Estimating nitrogen deficiency in wheat using RGB and multispectral UAV imagery. Remote Sens. **13**(12), 2345 (2021)
13. Tran, T.T., Choi, J.W., Le, T.T.H., Kim, J.W.: A comparative study of deep CNN in forecasting and classifying the macronutrient deficiencies on development of tomato plant. Appl. Sci. **9**(8), 1601 (2019)
14. Sethy, P.K., Barpanda, N.K., Rath, A.K., Behera, S.K.: Nitrogen deficiency prediction of rice crop based on convolutional neural network. J. Ambient Intell. Humaniz. Comput. **11**, 5703–5711 (2020)
15. Salaić, M., Novoselnik, F., Žarko, I.P., Galić, V.: Nitrogen deficiency in maize: annotated image classification dataset. Data Brief **50**, 109625 (2023)
16. Gul, Z., Bora, S.: Exploiting pre-trained convolutional neural networks for the detection of nutrient deficiencies in hydroponic basil. Sensors **23**(12), 5407 (2023)
17. Gupta, R., et al.: Hybrid methodology for nitrogen stress detection in rice plants using ResNet50 and U-Net. Comput. Electron. Agric. **182**, 105996 (2021)
18. Singh, A., et al.: Deep learning-based approach for detecting nitrogen deficiency in wheat crops. J. Plant Nutr. Soil Sci. **183**(2), 181–191 (2020)
19. Yu, F.H., Bai, J.C., Jin, Z.Y., Guo, Z.H., Yang, J.X., Chen, C.L.: Combining the critical nitrogen concentration and machine learning algorithms to estimate nitrogen deficiency in rice from UAV hyperspectral data. J. Integr. Agric. **22**(4), 1216–1229 (2023)

20. Juclà, D.G.I., Najdenovska, E., Dutoit, F., Raileanu, L.E.: Detecting stress caused by nitrogen deficit using deep learning techniques applied on plant electrophysiological data. Sci. Rep. **13**(1), 9633 (2023)
21. Wang, X., et al.: Image-based diagnosis of nitrogen deficiency in rice plants using deep learning models. Field Crops Res. **229**, 114–122 (2018)
22. Silva, C.A., et al.: Nitrogen deficiency diagnosis in sugarcane using convolutional neural networks. Sugar Tech **22**(5), 891–900 (2020)
23. https://www.kaggle.com/datasets/surangabandara/tea-leaves-for-nitrogen-deficiency

Ensemble-Based Classification of Bengali Crime News Headlines Using Machine Learning

Salman Islam, Md. Apu Hosen, Sk Fardeen Been Zaman, Rahatul Islam, Mohammad Nowsin Amin Sheikh, and Syed Md. Galib

Abstract This study presents a method for multi-class classification of crime-related Bengali newspaper headlines using ensemble machine learning techniques. The methodology begins with the development of a novel dataset categorized into eight distinct crime classes. Preprocessing involves TF-IDF-based bigram feature extraction to retain contextual meaning, followed by one-hot encoding of labels to ensure compatibility with machine learning models. Several base classifiers, including Logistic Regression, Random Forest, Naïve Bayes, and Support Vector Machine, were individually fine-tuned using Grid Search with cross-validation. A soft-voting ensemble model was then constructed, integrating the weighted outputs of the top-performing classifiers. The proposed ensemble approach achieved strong classification performance, demonstrating accuracy of 94%, precision of 94%, recall of 94%, and ROC AUC of 99%. Comparative analysis with existing methods confirms the superiority of this model in handling Bengali crime headline classification. This research contributes to the advancement of intelligent news analysis in Bengali and supports future developments in interpretable NLP applications for crime monitoring and public safety enhancement.

Keywords Bengali NLP · Bangla crime classification · Ensemble machine learning · Newspaper headline

S. Islam · Md. Apu Hosen · S. F. B. Zaman · R. Islam
Northern University of Business and Technology Khulna, Khulna, Bangladesh
e-mail: fardeen.zaman@icloud.com

Md. Apu Hosen · M. N. A. Sheikh (✉) · S. Md. Galib
Jashore University of Science and Technology, Jashore, Bangladesh
e-mail: n.amin@just.edu.bd

S. Md. Galib
e-mail: galib.cse@just.edu.bd

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_29

1 Introduction

Crime has remained a persistent societal challenge throughout history, prompting legal systems worldwide to categorize criminal acts for improved judicial procedures. In Bangladesh, a densely populated country with a growing economy, disparities in income, education, and limited law enforcement resources contribute to a relatively high crime rate. Given the widespread daily readership of newspapers in the country, organizing and classifying crime-related news in a systematic and automated manner becomes crucial for timely public awareness and policy planning.

Automatic text classification is a longstanding and significant research area in the field of natural language processing (NLP) and machine learning, especially with the proliferation of digital documents [1–3]. Broadly, text classification includes topic-based and genre-based approaches. Topic-based classification assigns documents to predefined categories based on their subject matter [4]. In the domain of public safety, automated analysis of crime and drug-related texts plays a vital role in enhancing crime prevention and investigative measures [5].

Ensemble learning is a powerful machine learning paradigm that combines the outputs of multiple base models to achieve better predictive performance than any individual model. A typical ensemble pipeline involves three key stages: feature extraction, deployment of multiple learning algorithms, and aggregation of results via adaptive or weighted voting mechanisms [6]. Since a document can be considered a sequence of words [7], it is usually represented as a feature vector of word occurrences. However, not all words are equally informative. Common words (stop words) are typically removed during preprocessing to improve model performance [8].

To address the unique linguistic and contextual nuances of Bengali-language crime reporting, specialized classification methods must be developed. Despite the growing body of NLP research in high-resource languages like English, Bengali remains underrepresented due to limited annotated datasets and linguistic tools. This lack of resources hinders the development of robust automated systems for content analysis, particularly in socially critical domains such as crime reporting. In response to this gap, this study proposes a multi-class classification framework tailored to Bengali crime-related newspaper headlines using ensemble machine learning techniques.

A novel dataset of Bengali crime headlines was developed, supporting the implementation of the proposed method. The approach leverages TF-IDF-based bigram feature extraction to preserve contextual relevance and employs a soft-voting ensemble strategy to integrate the strengths of multiple classifiers, including Logistic Regression, Random Forest, Naïve Bayes, and Support Vector Machine. The ensemble model achieved high performance, with 94% accuracy, precision, and recall, and an ROC AUC of 99%, outperforming existing methods. These findings highlight the effectiveness of ensemble learning in complex classification tasks and demonstrate the potential of automated systems for media monitoring, public safety planning, and digital content moderation in under-resourced languages like Bengali.

The rest of the paper is organized as follows: Sect. 2 presents a review of related work. Section 3 describes the proposed methodology. Section 4 discusses experimental results and analysis. Finally, Sect. 5 concludes the paper and suggests directions for future research.

2 Related Work

Many researchers have used various techniques to predict crime classification from a Bengali news portal. The earlier studies on this topic have been discussed in this section.

Tabashum et al. [9] investigated the effectiveness of different ML and DL models on a dataset of Bangla crime-related news. After applying preprocessing steps like tokenization, normalization, and stop-word removal, they used TF-IDF and Word2Vec for feature extraction. The models evaluated included Support Vector Machine (SVM), Naïve Bayes, Random Forest, Logistic Regression, Long Short-Term Memory (LSTM), and Bidirectional LSTM (Bi-LSTM). Among these, Bi-LSTM achieved the highest accuracy of 93.7%, while Random Forest performed best among traditional ML models. Similarly, Hossain et al. [10] addressed the classification of drug-related crime news using transformer-based deep learning models. They applied preprocessing steps similar to Tabashum et al. and used contextual embeddings suitable for transformers. Their models achieved an impressive accuracy of 94%, demonstrating their effectiveness in capturing nuanced linguistic features in Bangla crime news. Both studies highlight the growing potential of DL and transformer architectures in Bengali text classification tasks.

Khan et al. [11] developed a classification system for Bengali crime news headlines by categorizing 7872 samples into six types: terrorism, murder/attempt to murder, corruption, harassment, drug-related crimes, and snatching/stealing/robbery. Using data from sources like Kaggle and Prothom Alo, they applied standard preprocessing techniques such as null value removal, punctuation and stop-word elimination, and stemming with the Bangla Natural Language Processing Toolkit (BNLTK). Feature vectors were generated using TF-IDF, and a range of ML and DL models—including Logistic Regression, SVM, LSTM, Bi-LSTM, and Bangla-BERT—were tested. Bangla-BERT achieved the highest accuracy at 90.15%, while SVM attained 75.58%. Similarly, Khushbu et al. [12] proposed a multi-class classification system for 8602 Bengali news headlines spanning 11 categories. After preprocessing through tokenization, stop-word removal, and stemming, they experimented with various feature extraction techniques and trained ML models such as SVM, Naïve Bayes, and Random Forest alongside a neural network with an RNN encoder-decoder structure. This RNN-based model yielded the best result, achieving 90% accuracy. Both studies highlight the effectiveness of deep learning architectures, particularly BERT and RNNs, in Bengali text classification, especially when paired with robust preprocessing and feature extraction.

Rahman et al. [13] introduced a Graph Convolutional Network (GCN)-based approach for Bangla news classification across six domains: Education, Economy, International, Entertainment, Sports, and Technology. Their Text-GCN model was built using a graph of words and documents, linked via TF-IDF and PMI-based edges. Despite a small dataset sourced from the Prothom Alo portal, the model achieved an impressive 96.25% accuracy, outperforming BiLSTM, GRU-LSTM, LSTM, Char-CNN, and BERT in low-resource settings. Complementing this, Chy et al. [14] developed a Naïve Bayes classifier for multi-label categorization of Bangla news using the IPTC taxonomy. By leveraging a custom crawler and RSS parser, they collected and preprocessed news data, removing digits and punctuation, eliminating stop words with IDF, and applying stemming via a dictionary-based method. Their classifier, trained on 208 and tested on 416 manually labeled documents, achieved high accuracy across 34 categories. Together, these works emphasize that even lightweight models like Naïve Bayes or GCNs can deliver strong results when paired with domain-specific preprocessing and structured classification schemes.

Mahmud et al. [15] proposed an ensemble approach for classifying over 20,000 Bengali news articles into categories such as Sports, Politics, Entertainment, and Health. After applying traditional text preprocessing techniques, they compared ML and DL models, including SVM, Random Forest, LSTM, and GRU. To improve performance, they introduced an Election System Algorithm that selected the most frequent class predicted by multiple models. This ensemble strategy significantly boosted accuracy by 8.5% over individual models, achieving an overall classification accuracy of 95.45%. Their work demonstrates the power of hybrid modeling techniques and ensemble learning in improving classification robustness, especially in large-scale, multi-topic datasets.

Existing studies mainly focus on binary or broad multi-class classification of Bengali crime news. In contrast, our work introduces a detailed classification system across eight crime categories. Unlike prior research by Khan et al. and Hossain et al., we use a soft-voting ensemble of Logistic Regression, Naïve Bayes, Random Forest, and SVM to ensure both accuracy and efficiency in low-resource settings. We also provide a new labeled dataset and use bi-gram TF-IDF features to enhance context understanding without the high computational cost of deep learning models.

3 Methodology

This research proposes a multi-class classification framework to automatically categorize crime-related Bengali newspaper headlines using ensemble machine learning techniques. The methodology follows a systematic pipeline that includes data collection, preprocessing, feature extraction, model selection, hyperparameter tuning, and ensemble integration. Each step is designed to enhance the classification accuracy while preserving the semantic integrity of the headlines. Figure 1 illustrates the overall workflow of the proposed system, outlining the sequential stages from raw text input to final classification output.

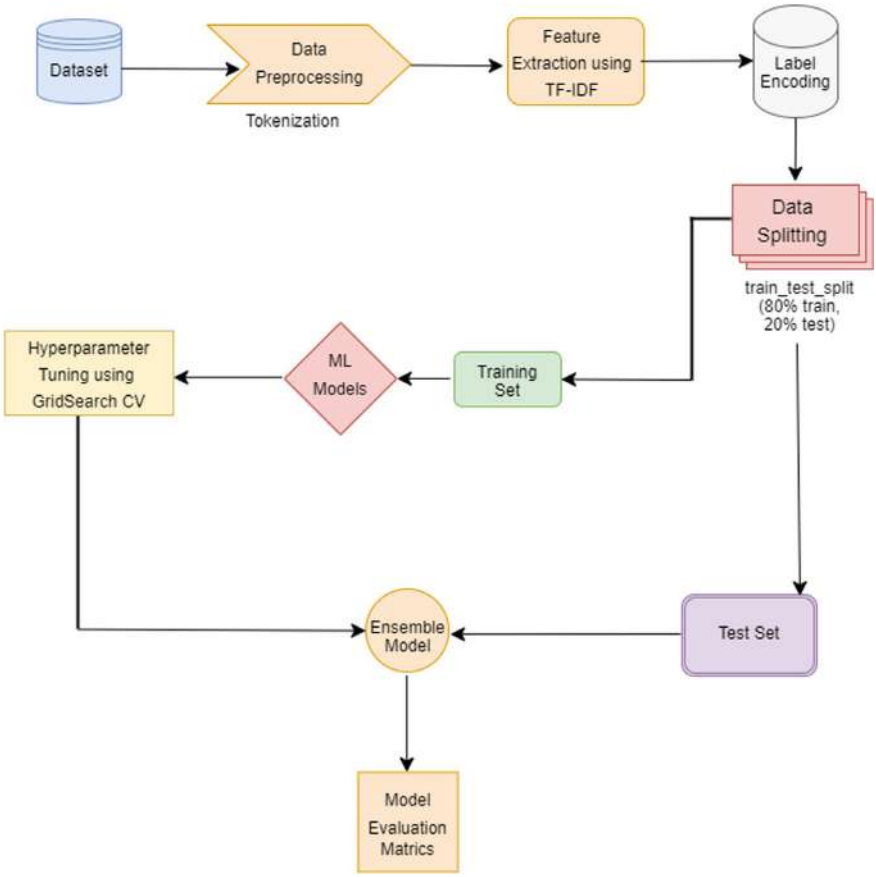


Fig. 1 Workflow of the proposed method

3.1 Data Collection

For this study, we introduced a dataset named "Bangla News Classification" dataset as shown in Table 1 aimed to facilitate crime text classification from Bengali news articles. The dataset contains 3000 data sourced from newspapers and categorized into 8 categories. It was then classified into eight categories in the dataset we gathered: rape, murder, robbery, attack, drug, kidnap, suicide, and theft as shown in Fig. 2. We obtained the target variable data based on the dataset we collected.

Table 1 Bangla news classification dataset

Headline	Category	Source
ঢাকার এলিফ্যান্ট রোডে ব্যবসায়ীকে কুপিয়ে…	Murder	https://www.prothomalo.com…
রাজধানীতে ডাকাতি ও ছিনতাইয়ের অভিযোগে গ্রেপ্তার ৭	Robbery	https://www.prothomalo.com…
রাজধানীতে ৪৫ ভরি স্বর্ণ চুরির অভিযোগে দুই কিশোর গ্রেপ্তার	Theft	https://www.prothomalo.com…
রাজধানীর মিরপুর থেকে অপহরণের ৭ ঘণ্টা পর ব্যবসায়ী উদ্ধার	Kidnap	https://www.prothomalo.com…
ঘরে ঢুকে গৃহবধূকে ধর্ষণের পর সঞ্চিত সব টাকাও নিয়ে গেল	Rape	https://www.prothomalo.com…

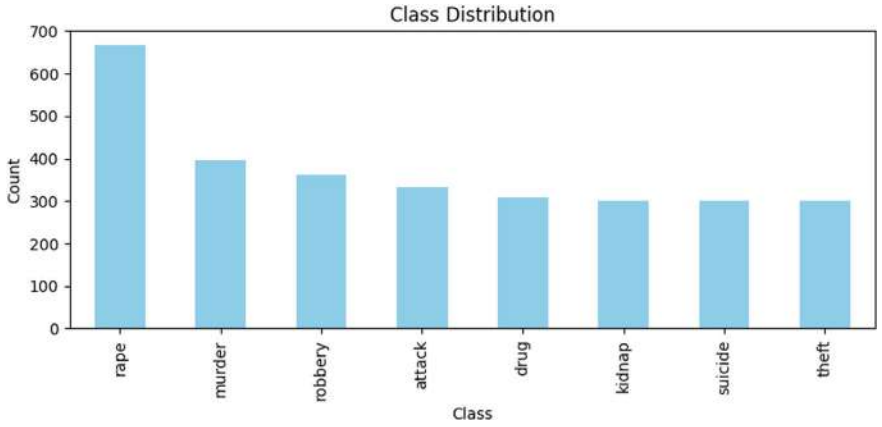


Fig. 2 Class distribution based on acquired data

3.2 Data Preprocessing

Data preprocessing is a crucial step in any machine learning pipeline, especially when working with textual data. To prepare the crime-related Bengali headlines for input into the classification models, several preprocessing operations were performed. Initially, all text was converted to lowercase, and unnecessary elements such as punctuation marks, special characters, and extra spaces were removed to ensure consistency and reduce noise in the data. Next, the text was transformed into numerical representations using Term Frequency-Inverse Document Frequency (TF-IDF), a widely used method that assigns greater importance to words that appear frequently in a specific document but less frequently across the entire dataset [16]. To capture contextual meaning more effectively, bi-grams [17] were used instead of individual

words. Bi-grams are pairs of consecutive words that help preserve phrase-level semantics, which is often important in crime-related language. Finally, the feature space was restricted to the top 5000 bi-grams based on their TF-IDF scores, ensuring a balance between model performance and computational efficiency.

3.3 Label Encoding

Label encoding is a fundamental preprocessing technique used to convert categorical variables into numerical form, enabling machine learning algorithms to process the data effectively [18]. In this step, the target labels (which indicate the category of each news headline) were converted into a numerical format so that they could be used in machine learning models. Since most machine learning algorithms cannot directly understand text labels like "murder" or "theft," we needed to transform them into numbers. To do this, we used LabelBinarizer, a tool from the Scikit-learn library. This method converts each category into a binary format, also known as one-hot encoding.

3.4 Hyperparameter Tuning

The dataset was divided into two parts: 80% for training and 20% for testing, ensuring a balanced evaluation of model performance. Several performance metrics were used to assess the classification models, including accuracy, precision, recall, F1-score, confusion matrix, Receiver Operating Characteristic (ROC) curve, and Area Under the Curve (AUC) [19]. To optimize model performance, hyperparameter tuning was conducted using Grid Search in combination with threefold cross-validation. This method systematically explores a predefined set of parameter values while evaluating the model on different subsets of the training data. Cross-validation reduces the risk of overfitting and helps ensure better generalization to unseen data. An ensemble model was used to enhance classification performance by combining the strengths of individual base classifiers. Hyperparameter tuning was also applied to the ensemble model to maximize its predictive accuracy. The specific hyperparameter configurations for each model are presented in Table 2.

3.5 Ensemble Model Creation

After evaluating the individual classifiers, an ensemble model was created to improve overall prediction performance. Specifically, we used a soft-voting ensemble classifier, which combines the output probabilities of multiple classifiers to make the final prediction. Unlike hard voting (which simply takes the majority class), soft

Table 2 Hyperparameters used in different models

Model	Hyperparameter	Value
Logistic Regression	C (regularization parameter)	1
	Max_iter	1000
Random Forest	n_estimators	200
Naïve Bayes	Alpha	0.5
SVM	Estimator	Linear SVC
Ensemble	Voting	Soft
	Weights	2, 2, 1, 2

voting considers the predicted probabilities from each model and chooses the class with the highest overall average probability. The ensemble was built using the best-tuned versions of the four models: Logistic Regression, Random Forest, Multinomial Naive Bayes, and SVM (with probability calibration). Each model's output was given a weight depending on how well it performed during validation.

3.6 Model Performance Evaluation

Evaluating the performance of machine learning models is a critical step in determining the effectiveness, reliability, and generalizability of a model. In this study, several widely accepted performance metrics have been employed, including accuracy, precision, recall, F1-score, confusion matrix, Receiver Operating Characteristic (ROC) curve, and Area Under the Curve (AUC). Each of these metrics provides a different perspective on how well the model performs, especially in the presence of imbalanced datasets or varying classification thresholds. These metrics collectively provide a comprehensive evaluation framework that ensures the model is not only accurate but also robust, especially under various data distributions and application scenarios. In this study, they have been used to compare and validate the predictive capabilities of different machine learning algorithms applied to the dataset.

4 Result and Discussion

The experiments were conducted using Google Colab Notebook, a cloud-based Jupyter environment that requires no local setup and provides free access to computational resources such as GPUs and TPUs. This platform facilitated rapid prototyping and iterative testing of machine learning models in an efficient and scalable manner. Essential Python libraries were used throughout the implementation, including Pandas for structured data manipulation, NumPy for efficient numerical

computations, and Matplotlib for generating visual representations of results and evaluation metrics.

All experiments were performed using Python 3, which provided extensive support for machine learning and natural language processing tasks. During execution, the environment utilized approximately 1.3 GB of system RAM (out of 12.7 GB available) and 32.7 GB of disk storage (out of 107.7 GB available), ensuring adequate computational resources for model training, evaluation, and result visualization. The cloud-based infrastructure not only enabled resource-efficient computation but also ensured reproducibility and accessibility for further experimentation and collaboration.

4.1 Classification Report

Ensemble-based methods demonstrated the most reliable and consistent performance across all models, showing strong capability in identifying subtle variations in Bengali crime-related headlines. These models achieved high scores in accuracy, precision, recall, F1-score, and ROC AUC, highlighting their effectiveness in handling complex classification tasks.

Among the individual classifiers, the Random Forest model showed well-balanced and robust results, confirming its suitability for moderately complex text classification. Support Vector Machine (SVM) also performed competitively, delivering high precision and recall. Both models proved effective in distinguishing between closely related crime categories. The ensemble model, which integrated the predictive strengths of top-performing classifiers, outperformed all standalone models. Its superior results confirm the benefits of ensemble learning, especially in text-based multi-class classification. A detailed comparison of model performance across various evaluation metrics is presented in Table 3, emphasizing the ensemble model's advantage in achieving greater classification accuracy and generalization.

Table 3 Classification report of different models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC (%)
LR	93	94	93	93	99
RF	93	94	93	93	98
NB	83	84	83	81	98
SVM	93	94	93	93	99
ENSEMBLE	94	94	94	94	99

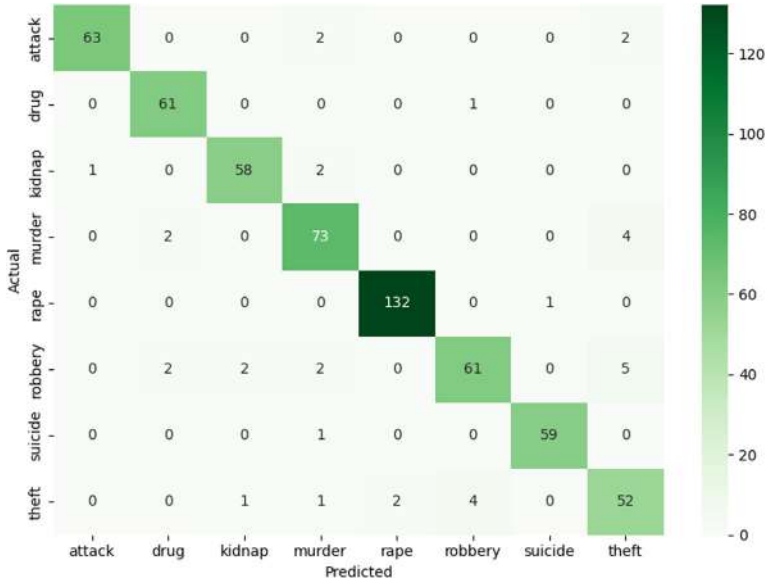


Fig. 3 Confusion matrix of ensemble model

4.2 Confusion Matrix

The confusion matrix shown in Fig. 3 provides deeper insights into the ensemble model's classification performance across eight crime categories. The model accurately classified the majority of instances in each class. For example, it correctly identified 132 out of 135 cases of rape, missing only 3 (2 false negatives and 1 misclassification). Similarly, it correctly classified 73 out of 77 murder cases, and 63 out of 67 attack cases, demonstrating strong performance in high-frequency categories. For less frequent crimes like kidnap and drug, the model maintained high precision, correctly classifying 58 out of 60 and 61 out of 63 cases, respectively. Minor misclassifications occurred across classes, such as robbery and theft, where a few instances overlapped. Overall, the matrix reflects the model's robust capability in handling multi-class classification with high accuracy, particularly for dominant classes, and minimal confusion among similar crime categories.

4.3 ROC Curve

The ROC curve illustrates the classification performance of the ensemble model across various crime categories, including attack, drug, kidnap, and others. Each colored line represents the Receiver Operating Characteristic (ROC) curve for a specific class. Curves that approach the top-left corner indicate better classification

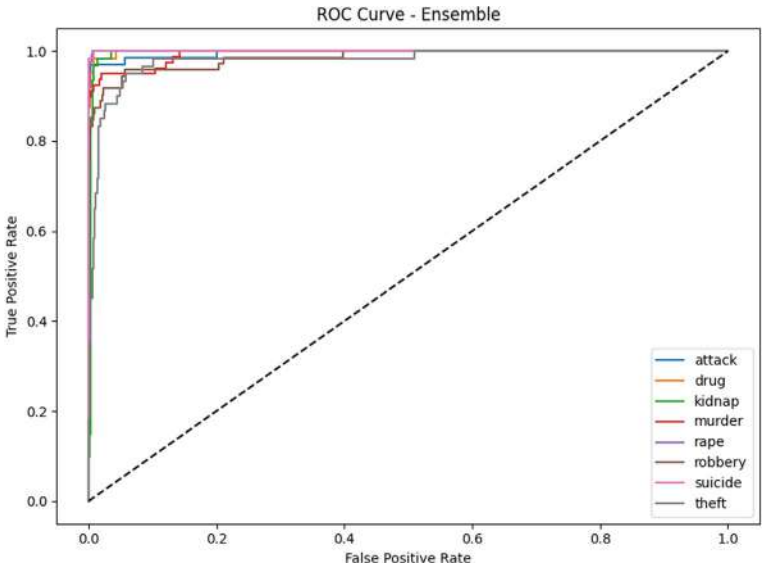


Fig. 4 ROC curve of the ensemble

performance, reflecting a high true positive rate and a low false positive rate. An Area Under the Curve (AUC) value of 0.99 demonstrates the model's excellent ability to distinguish between the different crime categories. As shown in Fig. 4, the ROC curves visually confirm the ensemble model's consistent and strong performance across all classes.

4.4 Comparative Analysis

After our analysis, we contrasted our model's performance with earlier research. However, once all the trials were finished, it was clear that our trained Ensemble model performed the best, as shown in Table 4. This demonstrates its higher ability to categorize academic achievement into classifying crime based on headlines.

5 Conclusion and Future Work

This study addresses the challenge of automatically classifying Bengali crime news headlines through a machine learning-based framework designed for fine-grained categorization. By introducing a new dataset comprising 3000 headlines labeled across eight distinct crime categories: rape, murder, robbery, attack, drug, kidnap,

Table 4 Performance comparison of proposed work with existing work

Work	Method	Category	Accuracy (%)	Precision (%)	Recall (%)	F1 score (%)
Tabashum et al. [9]	Shallow CNN	6	93	93	93	93
Hossain et al. [10]	Zero shot	4	94	96	92	94
Khan et al. [11]	Bangla-BERT	6	90	90	90	90
Proposed work	Ensemble	8	94	94	94	94

suicide, and theft, we fill a significant gap in existing research, which has predominantly focused on broader or binary classifications. Using TF-IDF-based bi-gram features and established machine learning models, we developed an ensemble system that integrates Logistic Regression, Naïve Bayes, Random Forest, and SVM classifiers through soft voting. This approach not only enhances classification performance but also maintains computational efficiency, making it suitable for real-world applications in low-resource environments. Our results demonstrate that traditional models, when optimized and combined effectively, can deliver competitive performance without the computational overhead of deep learning architectures. This work offers a practical and scalable solution for structured crime news analysis in the Bengali language, contributing meaningfully to both academic research and societal needs.

Future work includes expanding the dataset with additional news headlines from previous years and incorporating model interpretation techniques such as LIME for more comprehensive analysis. Given the limited amount of Natural Language Processing (NLP) research in the Bengali language, this study lays a strong foundation for future work in Bengali language processing.

References

1. Taha, K., Yoo, P.D., Yeun, C., Taha, A.: Text classification: a review, empirical, and experimental evaluation. arXiv preprint arXiv:2401.12982 (2024)
2. Zangari, A., Marcuzzo, M., Rizzo, M., Giudice, L., Albarelli, A., Gasparetto, A.: Hierarchical text classification and its foundations: a review of current research. Electronics **13**(7), 1199 (2024)
3. Fields, J., Chovanec, K., Madiraju, P.: A survey of text classification with transformers: how wide? How large? How long? How accurate? How expensive? How safe? IEEE Access **12**, 6518–6531 (2024)
4. Ozcalci, M., Kilic, M.: GA-LDA approach for topic modeling in Turkish accounting and finance articles: performance optimization in text classification. Spectr. Oper. Res. **2**(1), 305–322 (2025)

5. Mussiraliyeva, S., Baispay, G.: Leveraging machine learning methods for crime analysis in textual data. Int. J. Adv. Comput. Sci. Appl. **15**(4) (2024)
6. Yaghoubi, E., Yaghoubi, E., Khamees, A., Razmi, D., Lu, T.: A systematic review and meta-analysis of machine learning, deep learning, and ensemble learning approaches in predicting EV charging behavior. Eng. Appl. Artif. Intell. **135**, 108789 (2024)
7. Zulqarnain, M., Sheikh, R., Hussain, S., Sajid, M., Abbas, S.N., Majid, M., Ullah, U.: Text classification using deep learning models: a comparative review. Cloud Comput. Data Sci. 80–96 (2024)
8. Shukla, D., Dwivedi, S.K.: The study of the effect of preprocessing techniques for emotion detection on Amazon product review dataset. Soc. Netw. Anal. Min. **14**(1), 191 (2024)
9. Tabashum, S., Islam, A., Fami, F.N., Hossain, M.M., Zahara, M.Y.M.T.: Performance analysis of most prominent machine learning and deep learning algorithms in classifying Bangla crime news articles. In: 2020 IEEE 10 Symposium (TENSYMP) (2020)
10. Hossain, M.M., Chowdhury, Z.R., Akib, S.R.H., Ahmed, M.S., Hossain, M.M., Miah, A.S.M.: Crime text classification and drug modeling from Bengali news articles: a transformer network-based deep learning approach. In: 26th International Conference on Computer and Information Technology (ICCIT), Cox's Bazar, Bangladesh (2023)
11. Khan, N., Islam, M.S., Chowdhury, F., Siham, A.S., Sakib, M.N.: Bengali crime news classification based on newspaper headlines using NLP. In: 25th International Conference on Computer and Information Technology (ICCIT) (2022)
12. Khushbu, S.A., Masum, A.K.M., Abujar, S., Hossain, S.A.: Neural network based bengali news headline multi classification system: selection of feature describes comparative performance. In: 11th ICCCNT, Kharagpur (2020)
13. Rahman, M.M., Khan, M.A.Z., Biswas, A.A.: Bangla news classification using graph convolutional networks. In: International Conference on Computer Communication and Informatics (2021)
14. Chy, N., Seddiqui, M.H., Das, S.: Bangla news classification using Naive Bayes classifier. In: 16th International Conference Computer and Information Technology, Khulna, Bangladesh (2014)
15. Mahmud, T.A., Sultana, S., Mondal, A.: A new technique to classification of Bengali news grounded on ML and DL models. Int. J. Comput. Appl. **185**, 0975–8887 (2023)
16. Jones, K.S.: A statistical interpretation of term specificy and its application in retrieval. J. Document. **28**(1), 11–21
17. Hasanuzzaman, M., Hasan, M.R.: N-gram models for sentiment analysis of Bangla text. In: IEEE International Conference on Bangla Speech and Language Processing (2022)
18. Gong, J., Chen, T.: Does configuration encoding matter in learning software performance? An empirical study on encoding schemes. arXiv preprint (2022)
19. Ojajuni, O., Ayeni, F.: Predicting student academic performance using machine learning. In: Computational Science and Its Application, pp. 481–491 (2021)

IoT-Enabled Smart Belt and Mobile App for Enhancing Women's Safety

T. A. Mohanaprakash, D. R. Swathi Kumari, S. Nathiya, T. Sunitha, M. Therasa, and Manjunathan Alagarsamy

Abstract The alarming rise in women abuse and violence against women highlights the urgent need for smart, tech-driven safety solutions. This capstone project introduces a novel sensor-based system integrated with a smart waist belt and a mobile application to enhance real-time protection for women. The belt uses advanced sensors—such as accelerometers, and pressure sensors—to detect distress through sudden movements or abnormal pressure patterns. When triggered, the system sends an automatic alert containing the user's real-time GPS location and a distress message to pre-configured emergency contacts. A loud alarm can also be activated to attract attention and deter potential attackers. The mobile application, developed using Android studio, allows users to customize sensitivity levels, manage emergency contacts, and monitor alerts through a user-friendly interface. This integrated hardware-software solution aims to offer a reliable, scalable, and proactive tool for personal safety, empowering women and contributing to broader efforts in reducing violence and abuse.

T. A. Mohanaprakash (✉)
Department of CSE, RMK Engineering College, Chennai, India
e-mail: tamohanaprakash@gmail.com

D. R. Swathi Kumari
Department of CSE, SOET, CMR University, Bengaluru, India

S. Nathiya
Department of CSE, Excel Engineering College, Coimbatore, Tamil Nadu, India
e-mail: nathiyas.eec@excelcolleges.com

T. Sunitha
Department of Artificial Intelligence and Data Science, Saveetha Engineering College, Thandalam, Chennai, India

M. Therasa
Department of CSE, Panimalar Engineering College, Chennai, India

M. Alagarsamy
Department of Electronics and Communication Engineering, K. Ramakrishnan College of Technology, Trichy, Tamil Nadu, India

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_30

Keywords Women safety · Smart waist belt · Sensor-based system · Emergency alert · Real-time monitoring

1 Introduction

1.1 Background of the Project Domain

In recent years, the surge in sexual abuse and violence against women has emerged as a pervasive and urgent global concern. The issue is not limited to any one country or region but is prevalent worldwide, affecting women of all ages, ethnicities, and social backgrounds. Reports from organizations such as the World Health Organization (WHO) and the United Nations (UN) have shown that, on average, one in three women globally experiences some form of physical or sexual violence in her lifetime. This trend is particularly alarming in urban areas, where women are more likely to travel alone for work, education, or daily routines, making them vulnerable to potential threats.

Despite increased awareness and legal frameworks aimed at protecting women's rights, the frequency of sexual harassment, molestation, and abuse continues to rise. Societal efforts, including self-defense training and public awareness campaigns, have proven beneficial but insufficient in addressing the core issue of women's safety. In response, there has been growing interest in leveraging technological advancements to create more effective, reliable, and immediate solutions to protect women from harm.

1.2 Motivation and Relevance

The motivation behind this project is driven by the distressing reality that the number of crimes against women remains high despite various preventive measures taken by authorities and communities. Incidents of sexual violence not only compromise the physical safety of women but also inflict long-term psychological trauma, affecting their confidence, emotional well-being, and ability to engage in everyday activities without fear. Women often face restrictions on their mobility due to safety concerns, which in turn limits their professional opportunities and social interactions, ultimately contributing to gender inequality. Traditional safety solutions, such as wearable panic buttons or emergency calling apps, require manual activation by the user. However, during an abuse, the victim may not have the time, presence of mind, or physical ability to activate these safety features. This limitation highlights the need for automated, hands-free systems that can detect distress and send alerts autonomously. The proposed project seeks to bridge this gap by leveraging technology to enhance

women's safety through proactive measures that do not rely on user input during critical moments.

1.3 Problem Statement

Despite the rapid advancements in technology and the increasing proliferation of mobile safety applications, the world continues to witness alarming rates of violence against women. These incidents often occur in public spaces, homes, or other environments where victims may not have immediate access to help. Although there are numerous safety applications available today, many of them still require manual activation, making them impractical in situations where a victim might be unable to reach their device in time due to panic, physical incapacitation, or the suddenness of an abuse. In scenarios where every second counts, relying on a manual trigger can severely delay the response time, reducing the chances of timely intervention. Additionally, victims may not have the luxury of time or physical capacity to unlock their phones, navigate to an application, and press an emergency button. This highlights a fundamental limitation in current safety solutions, which are reactive rather than proactive. Given the inadequacy of traditional approaches to provide immediate and autonomous responses, there is a significant gap in the availability of solutions that can detect a distress situation automatically and initiate a real-time alert to emergency contacts without any manual intervention. For instance, existing safety applications can be useful under normal circumstances but may fail in scenarios where the victim is restrained, unconscious, or unable to access their phone due to physical or psychological shock.

Moreover, existing solutions do not fully leverage the advancements in wearable technology and IoT (Internet of Things) to create a seamless, integrated approach that works in real-time to protect women. The need for a wearable device that can detect, alert, and provide timely information in potential abuse situations is clear. Such a system could autonomously respond to an emergency situation, thereby significantly enhancing the chances of timely assistance and reducing the risk of harm to the victim. The problem this project seeks to address can be defined as follows.

"How can we develop a wearable device integrated with a mobile application that automatically detects and alerts trusted contacts during potential abuse situations, thereby enhancing the safety of women in real-time?" The solution to this problem lies in the development of a system that combines wearable technology with an intelligent, automated alert mechanism. By leveraging advanced sensors to detect unusual physical activities or impacts that might suggest distress, the system can autonomously send alerts to trusted contacts, significantly reducing response time. This project aims to fill the gap by providing a proactive and robust solution that can function even in extreme situations where the victim is unable to manually activate an SOS mechanism. The goal is to create a system that enhances personal safety, empowers women, and provides an extra layer of security during critical moments.

2 Related Works

Internet of Things (IoT) and Artificial Intelligence (AI) has facilitated the creation of real-time, intelligent safety solutions to cater to women's security and well-being. Recent studies identify several technological interventions that offer proactive and reactive support mechanisms.

Clement et al. [1] presented a smart wearable device with AI and IoT integration for real-time tracking and emergency assistance. The device integrates sensors and communication modules to identify distress and send notifications in real time to pre-identified contacts and authorities. The integration of GPS and biometrics ensures accurate location tracking and monitoring of health status, making the solution a comprehensive real-time safety system. Islam et al. [2] suggested an IoT framework for machine learning that detects possible harassment scenarios based on environmental and behavioral inputs. Through the use of AI models for data processing in real time, their system can recognize suspicious patterns of activity and autonomously trigger safety procedures. This highlights the potential use of predictive analytics for pre-emptive rather than reactive safety. Boomika et al. [3] proposed a GPS-based IoT tracking system designed specifically for women's safety. Their research aims at continuous location tracking and real-time communication, making sure that users are always traceable. The ease and convenience of the system make it ideal for actual deployment in various geographic and socio-economic environments. Khan and Amjad [4] reported an extensive review of current IoT-based women's safety solutions, comparing their efficacy, ease of use, and integration into larger smart city infrastructures. Their work highlights interoperability, user-oriented design, and privacy-enhancing mechanisms as crucial in such solutions.

Das and Swain [5] deployed an IoT module and cloud-based real-time safety system. Their system comprises panic buttons and built-in alerting features, providing prompt help and data storage for post-incident analysis. Although simpler compared to AI-based systems, the real-time nature of their model is especially important. Singh et al. [6] concentrated on the protection of IoT-enabled web applications employed within women's safety ecosystems. They promoted the deployment of sophisticated security protocols to secure sensitive user data and system integrity. Their work emphasizes the parallel challenge of security and safety while developing these solutions. Hadkar et al. [7] designed an effective IoT-based women's safety system prioritizing low power usage and light weight. Their system focuses on real-time alerting through GSM and GPS modules, and it is specially designed for deployment in locations where network coverage is limited.

Roy et al. [8] designed "Safe-Women," a smart safeguarding device that is IoT-enabled. The architecture uses GPS, GSM, and sensors to identify abrupt movements or falls and notify emergency contacts. The solution prioritizes the middle ground between price and technicality. Ponnusamy et al. [9] brought out a holistic volume outlining numerous AI, wearable, and surveillance technologies to support women's wellbeing. Their work amalgamates multiple views and applications such as monitoring, situational awareness, and psychological assistance by using AI-facilitated

systems. Finally, Kavitha et al. [10] investigated wearable IoT sensors for tracking women's health parameters, emphasizing the interconnectedness of physical health and personal safety. Their sensor monitors vital signs and alerts in case of any abnormalities, filling the gap between health tracking and security applications.

These studies collectively underscore the significant potential of integrating IoT and AI for women's safety. From real-time emergency responses to predictive threat detection and comprehensive monitoring, the literature emphasizes a multidisciplinary approach involving hardware design, software intelligence, and ethical data management. Future research should aim to enhance system scalability, ensure data privacy, and validate the real-world effectiveness of these technologies across diverse populations and environments.

3 Proposed System

The waist belt should incorporate a comprehensive set of advanced sensors, such as accelerometers, gyroscopes, and pressure sensors, to detect a range of potential distress signals. These sensors should be able to monitor the user's movements and identify any sudden shifts or abnormal pressure changes that could indicate a distress situation. The sensors should be capable of differentiating between normal physical activities and potentially harmful or violent motions, such as pushing, shoving, or sudden falls. The system should be capable of detecting small, nuanced changes in body movements to ensure no distress situation goes unnoticed. The system should be sensitive enough to identify various types of violent actions, including physical abuse or abrupt force exertion on the wearer. The sensors should be calibrated in a way that minimizes false positives, ensuring the device is activated only during genuine distress events. The integrated sensors should have real-time feedback capabilities, ensuring prompt detection and action when distress signals are detected.

Automatic Alerts

Upon detection of distress signals, the system should immediately initiate an automated SOS alert that includes critical details such as the user's current GPS location, timestamp, and a distress message. The system should be capable of notifying a predefined list of emergency contacts, including close family members, friends, and local emergency services. The system should also support the configuration of multiple emergency contacts to ensure a broader safety net in case one of them cannot respond. The SOS message should be transmitted through various communication channels, including SMS, push notifications, or even voice calls to ensure it reaches the emergency contacts as quickly as possible. The message should include the exact GPS coordinates, a live location link (e.g., Google Maps), and any relevant data, such as the type of distress detected (if applicable). The alert should be sent in such a way that it ensures the user's privacy while conveying the urgency of the situation. In case the GPS signal is weak or unavailable, the system should attempt to

use other location-detection methods, such as Wi-Fi triangulation or cellular tower information, to provide the most accurate location possible.

Emergency Alert System Integration

The system is designed to operate efficiently using hardware components and a Python-based alert mechanism, eliminating the need for a mobile application. Instead of relying on an app, the system uses Twilio API to send real-time SMS alerts to a registered mobile number whenever an emergency is detected. The entire process is handled by a Python script, which communicates with the Arduino-based hardware system to ensure immediate response in distress situations. The Arduino microcontroller continuously monitors inputs from the MPU6050 accelerometer and push button. If a sudden forceful movement is detected or the panic button is pressed, the Arduino triggers both the buzzer and a signal to the Python script running on a connected system (such as a laptop or Raspberry Pi). This script processes the alert and sends an emergency SMS to a predefined contact via Twilio API. To enhance reliability, the system allows users to configure emergency contacts directly in the Python script before deployment. This ensures that the alert message reaches the correct recipient instantly. Additionally, the system provides a manual override option through the push button, allowing the user to activate the alarm and send an alert even if the accelerometer does not detect movement. The buzzer plays a crucial role in the system by providing an immediate audio alert whenever an emergency is detected. It helps grab the attention of people nearby, increasing the chances of quick assistance. Since the system does not rely on a mobile app, the buzzer and SMS notification work independently, ensuring a robust and real-time response mechanism.

Loud Alarm

To ensure the device can effectively attract attention during an emergency, the waist belt should be equipped with a loud and attention-grabbing alarm that activates automatically when distress is detected. The alarm should be loud enough to be heard in noisy environments, ensuring bystanders are alerted to the user's distress. The loud alarm must be capable of sustaining its sound for a reasonable duration (at least 30–60 s) to ensure it is effective in attracting attention. Additionally, the alarm's tone should be distinct and difficult to ignore, ensuring its activation cannot be overlooked in public or crowded spaces. The alarm system should be designed to avoid accidental activations during non-distress situations, and its sound intensity should be adjustable through the mobile app in case the user needs to modify it for specific scenarios (e.g., quieter environments).

User Configuration

The mobile app should provide an intuitive user interface that allows the user to configure key settings easily. The app must include features that enable the user to input and modify emergency contacts, which will be notified in the event of a distress signal. The system should also offer options to adjust or deactivate certain features, such as the alarm sound or the frequency of updates. Additionally, users should be able to easily access historical data, such as a log of past alerts or distress signals,

within the app. A user-friendly onboarding process should be provided to guide new users through the setup and configuration process, ensuring they understand how to properly configure and use the device for maximum safety.

Non-functional Requirements-Reliability

The system must be highly reliable, ensuring that distress signals are accurately detected and alerts are sent out with minimal errors. False positives should be avoided as much as possible, with the system only triggering alarms in response to genuine distress situations. The system should be designed to operate under a wide range of environmental conditions, including varying temperatures, humidity levels, and physical conditions, without compromising on performance. It should also be able to function reliably across different devices and operating systems (iOS, Android). In cases where communication failures or technical issues may occur (e.g., Bluetooth disconnection or GPS failure), the system must be capable of either retrying the alert or switching to alternate communication protocols to ensure the alert still reaches the emergency contacts.

Power Considerations and System Reliability

As shown in Figs. 1 and 2, the women's safety smart belt operates entirely on a wired connection without incorporating any dedicated power source, such as a battery or rechargeable unit. Since the system does not require an internal power supply, its functionality depends on an external power source, such as a direct connection to a computer, power adapter, or any compatible supply via Arduino's USB interface. This setup ensures continuous operation as long as the system remains connected, making it suitable for real-time emergency alert functionality without concerns about battery depletion. Since no dedicated battery or power-saving mechanism is included, the system does not require additional power management. The Arduino, accelerometer, buzzer, and push button remain powered as long as the Arduino board itself is connected to an external power source. This eliminates the need for battery monitoring, charging circuits, or power optimization strategies, making the design simple and efficient. To maintain reliability, users must ensure that the Arduino is always properly connected to a power source when in use. If the system is deployed in a wearable format, a small portable power bank can be used to maintain its functionality without requiring access to a fixed power outlet. However, this is optional and depends on how the device is integrated into the final wearable form.

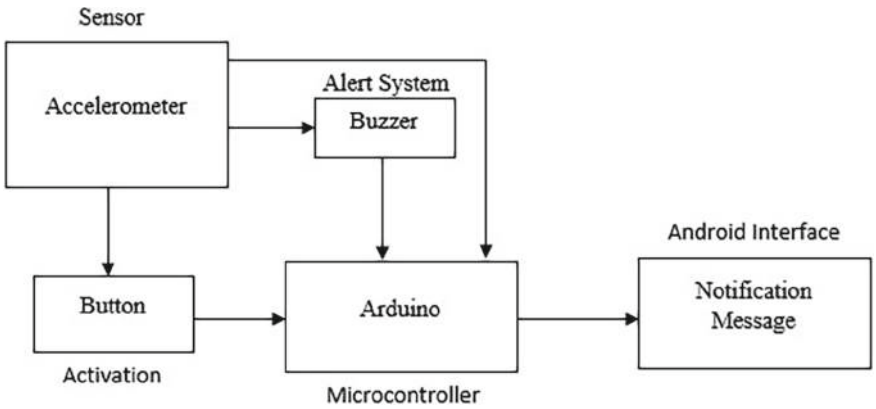


Fig. 1 Design architecture

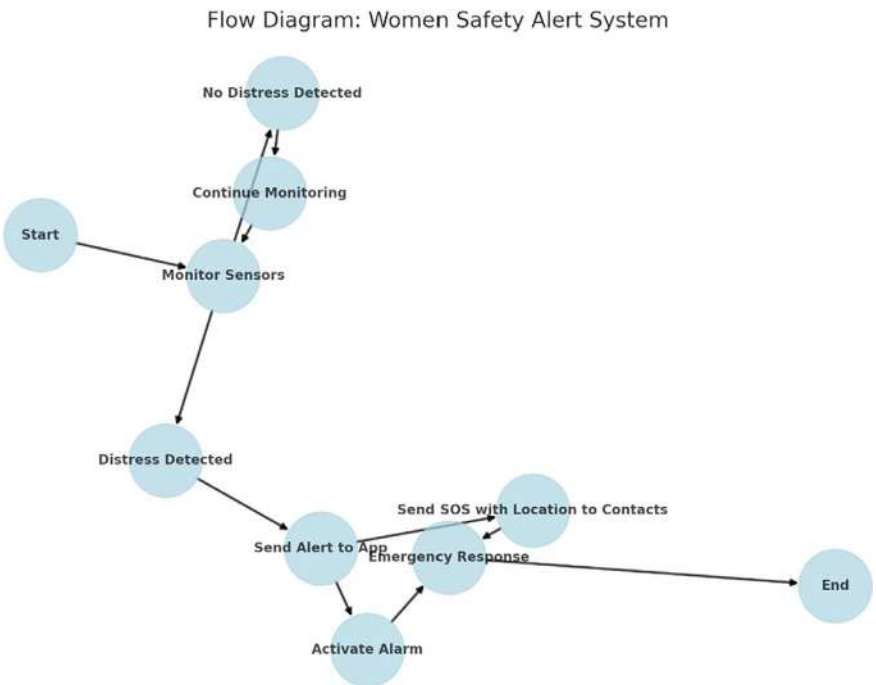


Fig. 2 Workflow diagram of the smart belt

4 Results and Discussion

4.1 Implementation

The Women's Safety Smart Belt is a wearable emergency alert system designed to enhance women's safety by detecting distress situations and notifying trusted contacts in real time. The system integrates an MPU6050 accelerometer, a push button, a buzzer, and an Arduino to create a compact and efficient safety solution. It operates in two ways: automatically and manually. The accelerometer continuously monitors movement, and if it detects sudden force or impact, the system triggers the buzzer and sends an emergency alert message to a registered mobile number via Twilio API. Additionally, the user can manually press the push button to activate the buzzer and send the alert message in case of an emergency. The system is designed to be small enough to fit inside a belt buckle, ensuring real-time usability and portability.

This smart belt provides an effective and immediate response to potential threats by combining sensor-based detection with manual activation. The buzzer acts as an immediate audio alarm to alert people nearby, discouraging attackers. The Arduino processes signals and communicates with a Python script, which sends an SMS alert to predefined contacts. Designed for low power consumption with a 3.8 V battery, the device ensures prolonged operation. Its compact and user-friendly design makes it a reliable and practical safety tool, offering a proactive approach to personal security in dangerous situations.

Working Principle

1. Sensor Activation: The MPU6050 accelerometer continuously monitors motion.
2. Sudden Force Detection: If a sudden force (above a threshold) is detected, it triggers an alert.
3. Manual Activation: The user can press a push button to manually trigger the alarm and send an emergency alert.
4. Buzzer Activation: The buzzer rings loudly to attract attention.
5. The message is delivered to the registered contact's mobile.

Arduino Code Implementation

```
#include <Wire.h>
#include <MPU6050.h>
MPU6050 mpu;
const int buttonPin = 2;   // Button connected to digital pin 2
const int buzzerPin = 3;   // Buzzer connected to digital pin 3
int lastState = HIGH;      // Track last button state
// Variables for movement detection
int16_t ax, ay, az, gx, gy, gz;
int16_t prevAx = 0, prevAy = 0, prevAz = 0;
const int threshold = 10000; // Adjust based on testing
void setup() {
```

```
    Serial.begin(9600);
    pinMode(buttonPin, INPUT_PULLUP);
    pinMode(buzzerPin, OUTPUT);
    // Initialize MPU6050
    Wire.begin();
    mpu.initialize();
    if (!mpu.testConnection()) {
      Serial.println("MPU6050 connection failed!");
      while (1);
    }
    Serial.println("MPU6050 connected.");
    delay(2000);
}
void loop() {
    int buttonState = digitalRead(buttonPin);
    // Button Press Logic
    if (buttonState == LOW && lastState == HIGH) {
      Serial.println("Button Pressed!");
      digitalWrite(buzzerPin, HIGH);
      delay(2000);
      digitalWrite(buzzerPin, LOW);
      delay(300);
    }
    lastState = buttonState;
    // Read Accelerometer Data
    mpu.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);
    int deltaX = abs(ax - prevAx);
    int deltaY = abs(ay - prevAy);
    int deltaZ = abs(az - prevAz);
    // Sudden Movement Detection
    if (deltaX > threshold || deltaY > threshold || deltaZ > threshold) {
      Serial.println(" 🚨Emergency! Sudden Movement Detected! 🚨");
    }
    // Print Sensor Data
    Serial.print("Accel (X, Y, Z): ");
    Serial.print(ax); Serial.print(", ");
    Serial.print(ay); Serial.print(", ");
    Serial.println(az);
    Serial.println("---------------------------");
    prevAx = ax;
    prevAy = ay;
    prevAz = az;
    delay(1000);
}
```

4.1.1 Python Script

```
Import serial
    import time
    from twilio.rest import Client
    SERIAL_PORT = "COM5"
    BAUD_RATE = 9600
    TWILIO_ACCOUNT_SID = "AC612233b816427583795903733f99bec5"
    TWILIO_AUTH_TOKEN = "c7c264845a4b2b1f69968cd4d1b9e0db"
    TWILIO_PHONE_NUMBER = "+17813588821"
    RECIPIENT_PHONE_NUMBER = "+919008436429"
    client = Client(TWILIO_ACCOUNT_SID, TWILIO_AUTH_TOKEN)
    def send_sms_alert():
        """Send an emergency alert SMS using Twilio."""
        message = client.messages.create(
            body=" Emergency Alert! Unusual Activity is Detected",
            from_=TWILIO_PHONE_NUMBER,
            to=RECIPIENT_PHONE_NUMBER
        )
        print(f"SMS sent! Message SID: {message.sid}")
    try:
        ser = serial.Serial(SERIAL_PORT, BAUD_RATE, timeout=1)
        time.sleep(2)
       print(f"Connected to {SERIAL_PORT}. Listening for emergency signals…\n")
        while True:
            line = ser.readline().decode('utf-8').strip()
            if line:
                print(line)
                    if "Button Pressed!" in line:
                    print(" ⚠Emergency detected! Sending SMS alert…")
                    send_sms_alert()
    except serial.SerialException:
        print("Failed to connect. Check if Arduino is properly connected & COM port
is correct.")
    except KeyboardInterrupt:
        print("\nExiting…")
        if ser:
            ser.close()
```

The Women's Safety Smart Belt shown in Fig. 3 incorporates an MPU6050 accelerometer to continuously monitor movement and detect sudden or unusual force. This sensor plays a crucial role in identifying distress situations, such as a push, hit, or fall. When the accelerometer detects a forceful movement that exceeds a predefined threshold, it automatically triggers the buzzer to produce a loud sound and sends an emergency alert message to a registered contact In addition to automatic

detection, the system includes a manual panic button (D2) that provides an alternative way to trigger an alert. If the user feels threatened or needs immediate help, they can press the button, which instantly activates the buzzer and sends an emergency SMS to a predefined mobile number. This feature is essential because not all distress situations involve a sudden force—some threats may require manual activation to seek help before escalation. The buzzer (D3) is a critical component in this system, serving as an audio alert mechanism. It is activated either by the accelerometer detecting forceful movement or by the user pressing the panic button. When activated, the buzzer produces a loud alarm, drawing the attention of nearby people and discouraging potential attackers. The buzzer ensures that even in situations where the mobile alert may take a few seconds to send, an immediate local response can be triggered to assist the user. The emergency message alert system is powered by Twilio, a cloud-based communication service that enables SMS messaging. When the Arduino detects an emergency, it sends a signal to a Python script, which runs on a computer or Raspberry Pi. The Python script processes the signal and triggers Twilio's API to send an SMS to the registered contact's phone number. This allows for a real-time communication system, ensuring that help is notified as quickly as possible. Overall, this system provides a multi-layered safety approach by combining automatic detection, manual activation, and real-time alerts. The accelerometer continuously monitors movement, the panic button allows user intervention, the buzzer provides an immediate local alert, and Twilio ensures remote assistance through emergency messaging. By integrating these elements into a compact belt buckle, the device remains discreet yet effective, making it a reliable and practical solution for women's safety in real-world scenarios.

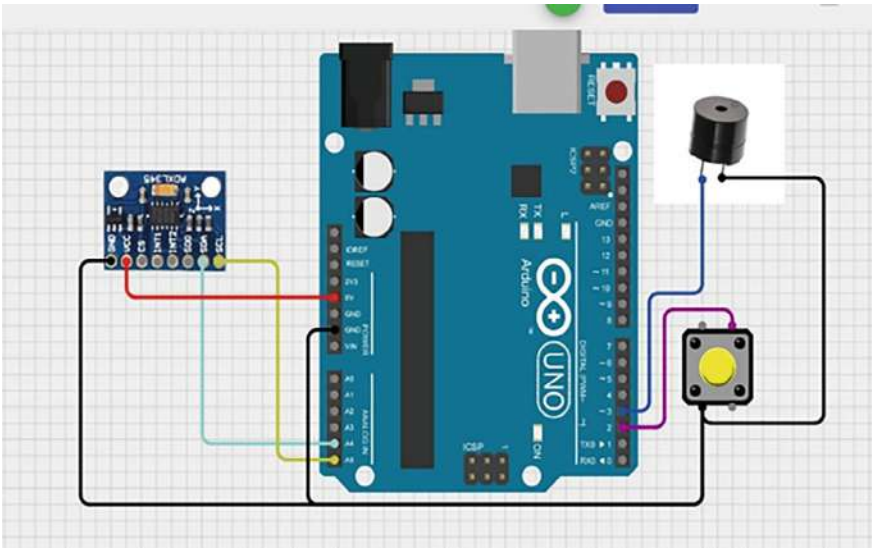


Fig. 3 Circuit diagram

4.2 Results

The women's safety smart belt system was successfully implemented and tested, demonstrating its effectiveness in detecting distress situations and providing an immediate emergency response. The system is designed to work with four key components: an MPU6050 accelerometer, an Arduino microcontroller, a buzzer, and a push button. These components work together to detect sudden forceful movements or manual distress signals, ensuring that an alert is triggered when necessary. During testing, the MPU6050 accelerometer effectively detected sudden jerks or forceful movements that might indicate an emergency situation. The Arduino processed these signals in real time and immediately activated the buzzer, providing an audible alert to attract attention. In addition to automatic detection, the push button served as a manual trigger, allowing the user to activate the alarm at any time in case of distress as shown in Fig. 4. This ensures that the system is not solely dependent on motion detection and can still function effectively in situations where movement may be restricted.

One of the key advantages of this system is that it does not require a separate mobile application or complex user interactions. The system operates independently and continuously monitors the wearer's safety without requiring manual intervention. The buzzer acts as an immediate distress signal, making it useful in situations where the user may not have access to their phone. The reliability of hardware-based alerts ensures that even if the user is unable to call for help manually, the system will automatically detect and respond to threats.

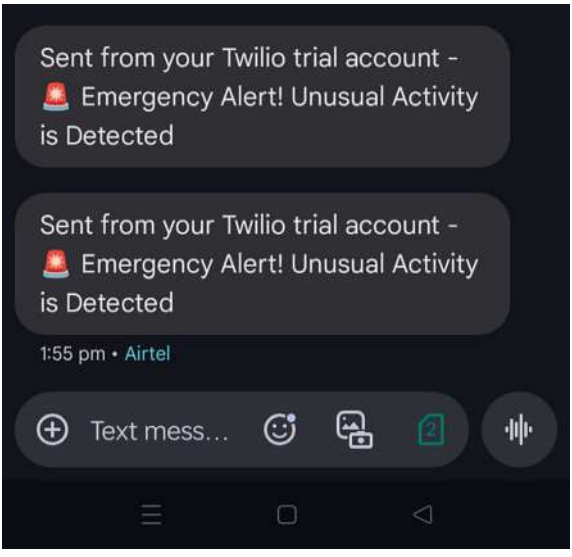


Fig. 4 Alert notification in mobile

The alert system was further enhanced by integrating a Python script that communicates with the Arduino. When either the accelerometer detects an abnormal movement or the button is pressed, the Arduino sends a signal to the Python script, which then uses Twilio's SMS API to send an emergency message to a registered contact number. This allows a predefined emergency contact to be notified instantly, ensuring that help can be sought as quickly as possible. The real-time response of the system was verified through multiple test scenarios, where alerts were successfully sent within seconds of an emergency trigger.

In conclusion, the implementation of the women's safety smart belt successfully meets the goal of real-time distress detection and emergency alert activation. Through a combination of motion sensing, manual triggering, and automated alert messaging, the system provides an effective and proactive safety mechanism. Its ability to function without dependence on external applications or continuous monitoring makes it a practical and reliable solution for real-world safety concerns.

5 Conclusıon and Future Scope

The women's safety smart belt has been successfully developed and tested, demonstrating its ability to provide a real-time distress detection and alert system. By integrating an MPU6050 accelerometer, Arduino microcontroller, buzzer, and a push button, the system effectively detects sudden forceful movements and provides a manual emergency activation option. The buzzer serves as an immediate alert mechanism, attracting the attention of people nearby, while the Python script linked to Twilio's SMS API ensures that emergency messages are promptly sent to a registered contact, enabling quick assistance in distress situations. Through rigorous testing, the system has proven to be highly responsive, accurate, and reliable. The accelerometer successfully detects unexpected forceful movements, triggering the necessary alerts, while the push button provides an additional layer of security, allowing users to manually send an alert if needed. The Python-based emergency messaging system ensures that critical notifications reach emergency contacts within seconds, enhancing the system's overall effectiveness in real-life scenarios.

A significant advantage of this system is its ability to function independently without requiring a mobile application. The hardware-based approach ensures that alerts are generated instantly, without requiring the user to access their phone or perform additional actions. The simplicity and real-time responsiveness of the system make it a practical and reliable safety solution for individuals facing dangerous situations. Furthermore, the design of this project eliminates complex dependencies on external software, making it an efficient and low-maintenance solution for personal safety. By focusing on sensor-based automatic detection and a manual emergency trigger, the system provides a comprehensive safety mechanism that can be integrated into wearable devices for real-world applications.

Additionally, the project highlights the importance of wearable safety technology in enhancing personal security. Since it operates on simple electronic components,

it is both cost-effective and scalable, making it accessible to a larger audience. The ability to detect emergencies autonomously and trigger alerts without requiring active user input ensures that the system remains effective even in situations where the wearer may be unable to call for help. With further refinements, such as size reduction using a PCB (Printed Circuit Board) instead of a breadboard, the system could be optimized for daily use, providing enhanced protection and peace of mind for individuals in vulnerable situations. In conclusion, the women's safety smart belt successfully addresses the need for an autonomous, fast, and reliable safety alert system. By combining sensor-driven detection, immediate audio alarms, and real-time emergency messaging, the system ensures that help can be sought quickly and efficiently in critical situations. Its ability to function without additional external dependencies makes it an ideal and practical safety solution for real-world implementation.

References

1. Clement, J., Ramya, R., Gomathi, T., Surendran, R., Kalyani, G.: Empowering women's safety: artificial intelligence and IoT-enabled smart wearable device for real-time monitoring and emergency assistance. In: 2025 8th International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech), pp. 1–5. IEEE (2025)
2. Islam, M.R., Oliullah, K., Kabir, M., Rahman, A., Mridha, M.F., Khan, M.F., Dey, N.: Machine learning-driven IoT device for women's safety: a real-time sexual harassment prevention system. Multimed. Tools Appl. 1–30 (2024)
3. Boomika, A., Divyapriya, E., Vanathi, K., Vidhya, N., Bhuvaneswari, P.T.: Empowering women safety: a GPS-enabled IoT tracking system. In: 2024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC), pp. 1–6. IEEE (2024)
4. Khan, R.N., Amjad, M.: Comprehensive evaluation of women's safety solutions in IoT environment. In: International Conference on Innovative Computing and Communication, pp. 323–355. Springer Nature Singapore, Singapore (2024)
5. Das, A.R., Swain, D.: IoT-based real-time women's safety system implementation
6. Singh, N.T., Dhiman, R., Yadav, A.L., Tanwar, H., Kumar, M., Kumar, G., Ruhela, A.K.: Securing IoT-enabled web applications and enhancing women's safety through advanced technologies. In: 2024 International Conference on Intelligent Systems for Cybersecurity (ISCS), pp. 1–5. IEEE (2024)
7. Hadkar, R.V., Phadke, A.P., Satpute, H., Raut, K., Deshpande, P.: An efficient IoT-enabled women safety device. In: 2024 3rd International Conference on Automation, Computing and Renewable Systems (ICACRS), pp. 425–431. IEEE (2024)
8. Roy, P., Noor, K.R., Shawon, S.M.: Safe-women: an IoT-enabled smart safeguarding device for enhancing women's security. In: 2024 IEEE International Conference on Power, Electrical, Electronics and Industrial Applications (PEEIACON), pp. 422–427. IEEE (2024)
9. Ponnusamy, S., Bora, V., Daigavane, P.M., Wazalwar, S.S. (eds.): Wearable Devices, Surveillance Systems, and AI for Women's Wellbeing. IGI Global (2024)
10. Kavitha, A., Aarthy, A., Dhanusha, R., Dhanusri, R.P., Indhuja, V.: Internet of things (IoT) based wearable device for monitoring women health. In: 2025 International Conference on Visual Analytics and Data Visualization (ICVADV), pp. 398–404. IEEE (2025)

Realtime Sign Language to Speech Conversion

Raj Bapat, Balasaheb Jadhav, Sanskar Kulkarni, Rriddhi Rathi, Sanyam Kothari, and Roshan Raut

Abstract Over 70 million deaf people worldwide rely on sign language for communication, enabling them to learn, work, and participate in society while fostering their inclusion within communities. However, teaching everyone sign language to ensure inclusivity and equal rights remains a significant challenge. This project aims to bridge this gap by developing a user-friendly human–computer interface (HCI) capable of understanding American Sign Language. By leveraging advanced tools like CNN, MediaPipe, and others, the system can process gestures and convert them into speech in real-time. This innovation not only simplifies communication but also empowers the deaf and mute communities, ensuring their voices are effectively heard and understood in day-to-day interactions.

Keywords Sign language · Real-time conversion · CNN · MediaPipe · Text-to-speech · Hand detection and tracking · Grouping and hierarchical classification · ASL · OpenCV

R. Bapat (✉) · B. Jadhav · S. Kulkarni · R. Rathi · S. Kothari · R. Raut
Department of Artificial Intelligence and Data Science (AI&DS), Vishwakarma Institute of Technology, Pune, India
e-mail: raj.bapat23@vit.edu

B. Jadhav
e-mail: balasaheb.jadhav@vit.edu

S. Kulkarni
e-mail: vilas.sanskar231@vit.edu

R. Rathi
e-mail: rriddhi.rathi23@vit.edu

S. Kothari
e-mail: sanyam.kothari23@vit.edu

R. Raut
e-mail: roshan.raut23@vit.edu

S. D. P. Ragavendiran et al. (eds.), *Innovations and Advances in Cognitive Systems*, Information Systems Engineering and Management 61,
https://doi.org/10.1007/978-3-031-97713-8_31

1 Introduction

Communication is a vital part of human interaction, yet over 70 million deaf and mute people worldwide face challenges when others cannot understand sign language. While sign language is effective within their community, its lack of universal understanding limits their inclusion in education, work, and society.

To bridge this gap, a real-time sign language-to-speech converter can transform communication by enabling seamless interaction between sign language users and non-users. This project focuses on developing a user-friendly human–computer interface that recognizes sign language gestures and converts them into speech. This project proposes a novel approach by utilizing a Convolutional Neural Network (CNN) and MediaPipe, which is a framework developed by Google used for building real-time multimedia applications and in this project is used to perform hand landmark detection.

The core contribution of this work to the landscape of already existing tools for real-time sign language to speech conversion is that this system identifies hand gestures with higher accuracy and speed, even in challenging conditions like poor lighting or complex backgrounds. Integrated with a text-to-speech tool, namely the Pyttsx3 library, the solution ensures real-time gesture recognition and speech synthesis for smooth and natural communication.

This innovation empowers the deaf and mute communities, enabling them to interact more effectively and fostering inclusivity in a world striving for accessibility and equality.

2 Literature Review

Considerable progress was seen in the last couple of years in developing the communication technologies to link the sign language users and their counterparts, who may or may not know sign language [1]. Such methods have included vision-based systems, wearable devices, and machine learning algorithms, followed by attempts at understanding sign language and communicating effectively [2].

Old systems for sign language recognition depended on hardware-intensive solutions, such as the glove-based devices wherein sensors would detect hand movements and gestures [3]. While accurate, the systems were costly, cumbersome, and impractical for real use across the land.

Thanks to the introduction of vision-based methods, the field was brought into revolution, looking not so natural and hardware-enabled. It uses computer vision techniques to detect and analyse gestures using standard cameras [4]. This reduces the need for additional equipment. However, vision-based approaches present their own obstacles, such as intensity changes caused by moving light sources, varying and complex background settings, different skin colour shades, all of which contribute to

lowering the accuracy [5]. Much of the research in operating with sign language translation is concerned with the vision-based approach. They focus upon recording the hand motion using a camera, followed by tracking the relevant features for recognition using computer vision. The affordable cameras and rapid developments of computer algorithms in the field have made the vision approach popular.

Recent developments in deep learning have been one of the major benefits for improving sign language recognition. Especially CNNs provide gesture recognition by learning spatial hierarchies and detecting complex patterns in an image. The immense power of CNNs demonstrated in the gesture classification of high accuracy has enabled robust sign language translation applications [6–8]. Other tools, like MediaPipe, have come into play as powerful real-time hand tracking and landmark detection. Where hand gestures can be tracked in difficult environments, MediaPipe really shines in those dynamic environments. Feedback loops integrated with CNNs and MediaPipe have built systems that can achieve incredible accuracy while working in less-than-the-best conditions.

While previous research, such as the work by [9] on Indian Sign Language translation, has demonstrated the feasibility of vision-based approaches, our work focuses specifically on American Sign Language and leverages the real-time capabilities of MediaPipe for robust hand tracking. Unlike the object detection approach employed by [10, 11], our system utilizes a CNN architecture specifically tailored for fine-grained gesture recognition based on the detailed hand landmarks provided by MediaPipe, potentially leading to improved accuracy in complex ASL gestures. Work by [8] mentions an accuracy of 86% which we have managed to increase to approx. 91% in challenging backgrounds and 97% in ideal conditions. The approach used by [12] uses a CNN and OpenCV for capturing the signs through a computer webcam but unlike our proposed method, it does not use MediaPipe for hand landmark detection. Using MediaPipe is a more efficient approach because it is specifically designed and optimized for tasks like hand tracking and the CNN can then learn to differentiate the hand signs based on these pre-processed images, reducing its complexity. Work by [13] talks about using a very similar approach as the one proposed in this paper with a combination of CNN and MediaPipe, however they mention the conversion of the recognized sign's to speech as a future scope, which we have managed to implement using the Pyttsx3 library.

Paper [14] talks about using transfer learning over the traditional CNN approach and showed an improvement of over 4% in accuracy. However, they used images with smooth backgrounds which is not possible to have in the real world and is something we have taken into consideration in our proposed method and shown we can maintain good accuracy.

Work by [15] looks into using various different algorithms to achieve the goal of real-time sign language to speech conversion and eventually finds the MediaPipe algorithm to be best suited and uses only that. However, their project did not talk about an easy-to-use user interface through a web app as proposed in our work. Our project is also focused on using a combination of CNN, MediaPipe and OpenCV to realize the goal of real-time sign language to speech conversion.

3 Methodology

The proposed real-time sign language-to-speech converter has the following components:

1. Acquire Dataset

Dataset sourced from Kaggle and was named "Indian Sign Language Dataset" by Soumya Kushwaha. The dataset contained on average 30 images for each letter in the alphabet. The dataset was licensed under 'CC0 1.0 Universal Deed', i.e., the dataset had no copyrights and the person who associated a work with this deed has dedicated the work to the public domain by waiving all of his or her rights to the work worldwide under copyright law, including all related and neighbouring rights, to the extent allowed by law.

2. Input Device

A standard webcam is used to capture live video frames of the user's hand gestures. This enables easy adaptation of the sign language to speech conversion by a large number of people.

3. Preprocessing Module

The preprocessing module carries out the following functionalities by using OpenCV.

(1) Cropping the Region of Interest (ROI)

This involves isolating just the area where the hand detected. By focusing only on the hand, we can eliminate irrelevant information such any objects in the background, making the subsequent steps faster and more accurate.

(2) Converting the Image to Grayscale

By removing color, we make the system more robust to variations in skin tone and lighting conditions.

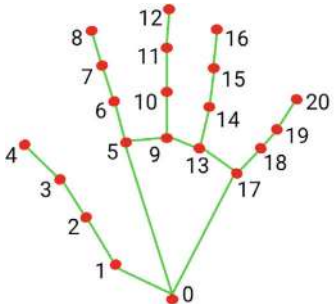
(3) Applying Gaussian Blur to Reduce Noise

This noise reduction step helps the program focus on the actual shape of the hand rather than any other factors that make prove to be a distraction. It makes the edges and overall form of the hand more distinct.

(4) Converting the Image into a Binary Format Using Thresholding Techniques

This creates a clear silhouette of the hand that emphasizes the hand's shape and boundaries. This is done by converting the previous grayscale image to black and white using a threshold value.

After carrying out these steps, the image is ready to be passed to the landmark detection phase.



0. WRIST
1. THUMB_CMC
2. THUMB_MCP
3. THUMB_IP
4. THUMB_TIP
5. INDEX_FINGER_MCP
6. INDEX_FINGER_PIP
7. INDEX_FINGER_DIP
8. INDEX_FINGER_TIP
9. MIDDLE_FINGER_MCP
10. MIDDLE_FINGER_PIP
11. MIDDLE_FINGER_DIP
12. MIDDLE_FINGER_TIP
13. RING_FINGER_MCP
14. RING_FINGER_PIP
15. RING_FINGER_DIP
16. RING_FINGER_TIP
17. PINKY_MCP
18. PINKY_PIP
19. PINKY_DIP
20. PINKY_TIP

Fig. 1 Landmarks detected by MediaPipe

4. Hand Detection and Tracking

The MediaPipe Library is employed to detect and track hand landmarks in real-time. By tracking the movement of hand landmarks, MediaPipe can accurately identify and classify different hand gestures. The landmarks MediaPipe detects are shown in Fig. 1.

5. Gesture Classification Model

A Convolutional Neural Network (CNN) is used to classify the pre-processed gestures. The CNN model is trained on labelled datasets of American Sign Language (ASL) gestures. The dataset contained on an average 30 image files for each alphabet.

6. Grouping and Hierarchical Classification

Gestures are grouped into logical classes to improve accuracy. Initially, the extracted features of the pre-processed hand gestures are subjected to a rough classification, wherein gestures exhibiting similar visual characteristics are clustered into discrete, logically classes. This initial partitioning reduces the dimensionality of the classification space for subsequent stages. For instance, similar-looking gestures are classified into a group before further distinction is made within the group.

Subsequently, a more through classification is performed within each identified group. This stage employs more discriminative features to differentiate between gestures sharing the initially identified rough attributes.

7. Text Output

Once the CNN model has successfully identified the letter, it can output it in text format. Our project involves an additional sign to signify the saving of the letter. Using this, the user can form words which then can be converted to speech.

8. Text-to-Speech Conversion

The Pyttsx3 Library is utilized to transform the text output into speech, providing auditory feedback in real-time.

9. User Interface

A simple and interactive graphical user interface (GUI) displays the recognized gestures as text and provides speech output for seamless communication. It also provides suggestions on what the word could be as the user is signing the letters to enhance efficiency.

By combining these components, the system ensures efficient, accurate, and user-friendly translation of sign language gestures into speech, addressing real-world challenges like varying lighting conditions and complex backgrounds.

Figure 2 shows our proposed algorithm and visually represents the flow of our proposed system.

The novelty of our approach lies in the integration of a dual-output architecture that simultaneously translates sign language into both text and speech using a unified neural network-based image processing system. Unlike traditional systems that focus solely on either text conversion or require predefined gesture datasets, our model is trained on dynamic image sequences, enabling it to recognize continuous gestures in real time. Additionally, the inclusion of a lightweight speech synthesis layer makes the system more accessible for real-world applications, particularly for

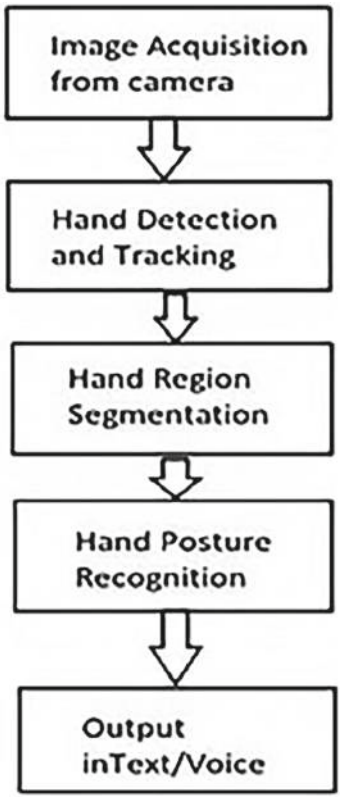


Fig. 2 Flowchart of algorithm

assisting individuals with speech or hearing impairments in everyday communication. The proposed pipeline also emphasizes low-latency processing, which is crucial for interactive use cases—a feature that sets it apart from many existing solutions which suffer from lag or require post-processing.

3.1 System Algorithm

The system begins by initializing the necessary libraries: MediaPipe for hand detection, OpenCV for image preprocessing, a pre-trained Convolutional Neural Network (CNN) model for gesture classification, and Pyttsx3 for text-to-speech conversion. Once the libraries are set up, the webcam is activated to capture a continuous stream of video frames. These frames are read one by one for pre-processing.

The captured video frame is then analysed to detect the presence of a hand and the Region of Interest (ROI) containing the hand is extracted. The hand image is then pre-processed by cropping the ROI, converting the image to grayscale, applying Gaussian blur to reduce noise, and finally converting the grayscale image into a binary image using thresholding techniques. Then MediaPipe is used to identify the hand landmarks.

After pre-processing, the cropped and processed image is fed into the pre-trained CNN model for gesture classification. The CNN model classifies the hand gesture into one of the predefined categories, which could be individual alphabets or grouped classes. In cases where the gesture belongs to a grouped class, the system further distinguishes between similar gestures based on the hand landmarks to ensure accurate classification.

Once the gesture is identified, it is mapped to the corresponding text, which represents the recognized alphabet. This text is displayed on the graphical user interface (GUI), and Pyttsx3 is used to convert the text into speech for real-time auditory feedback. The system continues to process and display new gestures as they are recognized, providing a continuous cycle of gesture detection, classification, and conversion to speech. The process continues until the user decides to terminate the system, after which the webcam is deactivated, and all resources are released.

4 Results and Discussions

The proposed solution was selected based on the increasing need for an inclusive and seamless communication bridge between hearing-impaired individuals and the general public. Neural networks, particularly convolutional and recurrent architectures, have demonstrated superior performance in pattern recognition tasks such as image classification and sequence prediction. Given the visual and temporal nature of sign language, a neural network-based approach is well-suited for capturing both spatial hand movements and their temporal context. MediaPipe was selected due

to its ready-made hand landmark detection functionality. This enabled us to train the CNN on these landmarks instead of the entire images thus helping us improve efficiency by keeping the complexity of the CNN to a minimum. Furthermore, integrating this with a speech and text output system ensures that the communication is not only interpreted correctly but also delivered in a form that is easily understandable by the listener. This comprehensive design addresses the limitations of earlier methods, which often relied on static datasets, required specialized hardware, or lacked real-time adaptability.

Figure 3 shows the User-Interface of our project. The user can see the signed characters and also gets suggestions to complete the word efficiently. 'Clear' and 'Speak' buttons are also given to clear the letters and convert the signed letters to speech respectively.

In controlled environments with good lighting and clean backgrounds, the system performed exceptionally well, reaching up to 97% accuracy. In condition scenarios whose variables were real-life situations such as fluctuating light intensity, complex background, or hand occlusion, the performance exhibited a slight drop in accuracy with the accuracy becoming 91%. This was mainly due to the difficulties related to hand detection and landmark accuracy because the system was designed to rely on clearly visible and well-lit images for accurate detection and tracking of hand gestures. We believe if the CNN model is trained on more challenging images, the accuracy in challenging conditions may improve.

The real-time gesture recognition and conversion into speech has been achieved without significant delays. The system processes the video frames and provides immediate feedback by displaying the recognized alphabet as text and converting it into speech once the word is complete. The user interface is responsive, and since the system also shows the hand landmarks detected, the user can get suggestions on how

Fig. 3 Implementation and user-interface of proposed system

to position their hand more effectively for optimum detection. The use of MediaPipe for hand tracking and OpenCV for image preprocessing ensures smooth operation even with moderate computational resources.

While the system demonstrated impressive results, certain limitations need to be addressed for broader applicability. One of the key challenges faced was the system's dependency on ideal background and lighting conditions. In scenarios where there is inconsistent lighting or a cluttered background, the accuracy of hand detection and gesture recognition was compromised. Additionally, real-time gesture recognition still struggles with more complex gestures and especially fast hand movements, which could lead to missed or incorrect classifications. The system also needs further refinement to handle regional variations in sign language. Many regions may have dialectical differences or specific gestures that are not recognized by the model, limiting its universal applicability. The categorization of 26 alphabets into 8 classes helps in improving the system's accuracy, but future work should focus on expanding the model to recognize complete words and phrases which will help in the project's deployment in the real world.

5 Conclusion

In conclusion, the suggested sign language-to-speech system is an extremely efficient and universal real-time gesture recognition solution. With the use of advanced methods like OpenCV for preprocessing, MediaPipe for hand landmark detection and CNN for gesture recognition, the system gives timely and accurate feedback, translating sign language gestures to speech with little delay.

The primary contribution of this study is the close to 91% accuracy found after testing in difficult conditions, including noisy backgrounds and low lighting, which emphasizes its robustness for real-world applications. Under ideal conditions, the system has 97% accuracy. This shows that our proposed methodology and system can be more accurate and faster than existing systems.

Although there are some limitations, including reliance on ideal backgrounds and lighting, the system's performance lays a strong basis for future enhancement. Enlargement of the dataset, added robustness, and addition of regional sign language variations are promising directions for future enhancement, ensuring the system's enhanced applicability and inclusivity. This work is a significant step towards filling communication gaps for the deaf and hard-of-hearing population, making it a useful tool for hands-free real-time communication.

Acknowledgements The photos included in the implementation section show one of our co-authors Mr. Sanskar Kulkarni. His involvement in the pictures provided for publication was only included after receiving written informed consent prior to his involvement. We thank him for his cooperation.

We would also like to extend our deepest gratitude to our project guide, Prof. Balasaheb Jadhav, whose mentorship and expert advice were crucial in shaping the project's vision and overcoming challenges. We also appreciate Vishwakarma Institute of Technology for providing a supportive

environment that fostered learning and innovation. The institute's resources and academic backing were essential for the successful development and implementation of the project.

References

1. Yousaf, K., Mehmood, Z., Saba, T., Rehman, A., Rashid, M., Altaf, M., Shuguang, Z.: A novel technique for speech recognition and visualization based mobile application to support two-way communication between deaf-mute and normal peoples. Wireless Commun. Mob. Comput. (2018)
2. Madhiarasan, M., Roy, P.P.: A comprehensive review of sign language recognition: different types, modalities, and datasets, 7 Apr 2022
3. Pezzuoli, F., Corona, D., Corradini, M.L.D.: Recognition and classification of dynamic hand gestures by a wearable data-glove. SN Comput. Sci. (2020)
4. Shah, P., Pandya, K., Shah, H., Gandhi, J.: Survey on vision based hand gesture recognition. Int. J. Comput. Sci. Eng. **7**(5) (2019)
5. Rautaray, S.S., Agrawal, A.: Vision based hand gesture recognition for human computer interaction: a survey. Artif. Intell. Rev. **43**, 1–54 (2012)
6. Gadekallu, T.R., Alazab, M., Kaluri, R., Maddikunta, P.K.R., Bhattacharya, S., Lakshmanna, K.: Hand gesture classification using a novel CNN-crow search algorithm. Complex Intell. Syst. **7**, 1855–1868 (2021)
7. Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C.L., Grundmann, M.: MediaPipe hands: on-device real-time hand tracking (2020). [Online]. Available: https://arxiv.org/abs/2006.10214
8. Repal, P.: Real time sign language translator using machine learning. J. Artif. Intell. Mach. Learn. Neural Netw. **4** (2024)
9. Doshi, S., Joshi, R., Chavan, S., Burli, P., Kulkarni, S.: Vision-based real-time Indian sign language translator. J. Emerg. Technol. Innov. Res. **6**(3) (2019)
10. Verma, P., Badli, K.: Real-time sign language detection using TensorFlow, OpenCV and Python. Int. J. Res. Appl. Sci. Eng. Technol. (2022)
11. Pathak, A., Kumar, A., Priyam, Gupta, P., Chugh, G.: Real time sign language detection. Int. J. Mod. Trends Sci. Technol. (2021)
12. Ojha, A., Pandey, A., Maurya, S., Thakur, A., Dayananda, P.: Sign language to text and speech translation in real time using convolutional neural network. In: National Conference on Advancements in Information Technology, vol. 8, no. 15 (2020)
13. Verma, A.R., Singh, G., Meghwal, K., Ramji, B., Dadheech, P.K.: Enhancing sign language detection through MediaPipe and convolutional neural networks (CNN) (2024)
14. Thakar, S., Shah, S., Shah, B., Nimkar, A.V.: Sign language to text conversion in real time using transfer learning. In: 3rd International Conference for Advancement in Technology (ICONAT) (2024)
15. Prabhakar, M., Hundekar, P., Sai Deepthi, B.P., Tiwari, S., Vinutha, M.S.: Sign language conversion to text and speech. J. Emerg. Technol. Innov. Res. **9**(7) (2022)