

Fraud Transaction Detection Using Machine Learning

Niranjan Sathe¹, Tushar Yeola², Aryan Shete³, Prof. Priyanka Shahane⁴

¹Student, SCTR's Pune Institute of Computer Technology, Pune.

²Student, SCTR's Pune Institute of Computer Technology, Pune

³Student, SCTR's Pune Institute of Computer Technology, Pune

⁴Assistant Professor, SCTR's Pune Institute of Computer Technology, Pune.

Abstract - The rapid growth of digital banking, e-commerce, and online payment systems has significantly increased fraudulent activities, resulting in financial losses and reduced customer trust. Traditional fraud detection approaches often struggle with evolving fraud patterns, data imbalance, and false positives, creating the need for more robust and efficient solutions. This project proposes a machine learning-based fraud transaction detection framework that utilizes supervised learning algorithms, specifically Logistic Regression and Random Forest, for identifying fraudulent transactions.

The system focuses on data preprocessing, input validation, and feature engineering to improve data quality and enhance model performance. Transaction data is divided into training and testing sets to build predictive models capable of learning patterns associated with fraudulent behavior. A risk evaluation engine analyzes model predictions to assign transaction risk levels, while an alert generation module provides structured alerts for suspicious activities. The system further integrates a dashboard interface for visualizing fraud alerts and transaction analysis.

The proposed framework is evaluated using standard classification metrics such as Accuracy, Precision, Recall, and F1-score to assess detection performance. By combining machine learning models with structured preprocessing, risk evaluation, and alert generation, the system provides an accurate, scalable, and computationally efficient fraud detection solution suitable for financial transaction monitoring and real-time fraud prevention.

Key Words: Fraud Detection, Machine Learning, Online Payment Security, Logistic Regression, Random Forest, Performance Evaluation, Financial Risk Assessment.

1.INTRODUCTION

The rapid digital transformation of financial ecosystems, driven by the widespread adoption of online banking, mobile payment systems, and e-commerce platforms, has fundamentally reshaped transactional conduct. While cashless economies have enhanced speed and convenience, they have simultaneously expanded the attack surface for cybercriminals. This has led to a substantial rise in sophisticated financial fraud schemes, including identity theft, account takeovers, and card-not-present fraud. As modern financial institutions process millions of transactions at high speed, the scale and velocity of transactional data render manual auditing infeasible, necessitating the deployment of automated and intelligent fraud detection mechanisms.

Traditional rule-based systems suffer from inherent rigidity and high false-positive rates. Static detection approaches often fail to adapt to evolving fraud patterns, causing model performance

to degrade over time. To maintain high detection rates, there is a critical need for a machine learning perspective that prioritizes model adaptability, predictive precision, and efficient processing. Effective fraud detection requires moving beyond simple threshold-based monitoring toward robust algorithms capable of identifying subtle patterns hidden within large transactional datasets.

Existing fraud detection systems face multiple challenges. First, financial transaction datasets are highly imbalanced, where fraudulent transactions represent only a small fraction of the total data, leading to biased learning behavior and reduced model generalization. Second, real-time transaction monitoring requires models that are not only accurate but also computationally efficient. Third, reducing false positives while maintaining strong fraud detection capability is essential for operational reliability and user trust. Addressing these challenges requires rigorous data preprocessing, feature engineering, and the comparative evaluation of supervised machine learning models to identify the most reliable solution.

To address these limitations, this project proposes an intelligent fraud transaction detection framework based on a machine learning approach. The proposed system evaluates the effectiveness of supervised learning algorithms, specifically Logistic Regression and Random Forest, for detecting fraudulent transactions. The framework incorporates input validation, feature engineering, train-test data splitting, model training, and fraud prediction to build a structured detection pipeline. A risk evaluation engine analyzes model outputs to identify suspicious transactions, while an alert generation module provides structured fraud alerts for monitoring purposes.

Furthermore, the framework is designed to ensure operational scalability and effective fraud detection performance for practical financial applications. By integrating machine learning models with robust preprocessing, transaction risk evaluation, and dashboard-based visualization, the system aims to reduce fraudulent activities and support real-time monitoring of suspicious behavior. This provides a scalable and practical fraud detection solution suitable for modern digital financial ecosystems and establishes a foundation for future enhancements in intelligent financial security systems.

2. LITERATURE SURVEY

Recent advancements in machine learning and deep learning have significantly enhanced fraud detection systems in digital financial environments. Early research primarily focused on supervised learning models for transaction classification using structured features. However, with the increasing complexity of fraud patterns, modern studies have shifted towards specialized behavior modeling, graph-based representations, and privacy-preserving distributed systems. While these advanced methods offer unique advantages, a critical research direction emphasizes the "Machine Learning Perspective," focusing on the optimization of established supervised algorithms to achieve superior accuracy and computational efficiency.

Table-1: Comparative Analysis of Fraud Detection Techniques in Online Payment Systems

Sr. No.	Paper	Best Classification Technique	Dataset	Performance Parameters	Features
1	ML Perspective: Fraud Payment Detection [1]	Logistic Regression	Real-world payments	Acc, Prec, Rec, F1	Comparative ML analysis for efficiency
2	Behaviour Tree with LIC [2]	LIC Tree-LSTM	Alibaba dataset	Acc, Prec, Rec, AUC	Sequential behavior modeling
3	Ranking Metric Embedding-Based Multicontextual Behaviour Profiling[3]	Embedding-based	Banking transactions	Prec, Rec, FPR	Multi-context behavior profiling
4	Quantum ML Fraud Detection [4]	Quantum SVM	Credit + loan data	Acc, Speed, Prec	Quantum optimization for imbalance
5	Hybrid Deep Learning Model [5]	ResNeXt + GRU	Financial datasets	Acc, Prec, Rec, F1, AUC	CNN + RNN hybrid modeling
6	GNN + Autoencoder [6]	GNN + AE	Banking data	Prec, Rec, Acc	Graph-based anomaly detection
7	Spatial-Temporal GNN [7]	ST-GNN	Yelp, Amazon	AUC, F1, G-mean	Spatio-temporal fraud patterns
8	Drift Detection Framework [8]	CNN + ADWIN	Mixed datasets	Acc, Prec, Drift	Adaptive drift detection

9	FedShield FL Model [9]	Federated Learning	Distributed data	F1, Prec, Rec	Privacy-preserving detection
10	FraudX SimS Dataset [10]	Random Forest	Synthetic dataset	Acc, Prec, Rec, AUC	Synthetic data for evaluation

Providing a critical benchmark for practical application, Upadhyay et al. [1] analyzed fraud detection from a systematic perspective, evaluating the effectiveness of Logistic Regression, Random Forest, and XGBoost on online payment transaction data. Their research demonstrated that by optimizing these supervised learning algorithms and focusing on rigorous feature selection, a superior accuracy of 99.51% could be achieved. This study underscores that well-tuned traditional machine learning models often provide a more reliable and efficient solution for real-time systems compared to high-dimensional neural architectures, provided that data preprocessing is handled rigorously.

In contrast to the broad machine learning perspective, Liu et al. [2] proposed a specialized fraud detection framework based on Behaviour Trees integrated with Local Intention Calibration (LIC Tree-LSTM). Their approach models user behavioral sequences and captures hidden transactional intentions using neural architectures. While the LIC Tree-LSTM outperformed traditional sequential models like RNN and LSTM on e-commerce datasets, its high computational complexity and focus on sequential patterns often exclude the relational dependencies across broader entities.

To enhance contextual profiling, Cui et al. [3] introduced the ReMEMBER framework, a Ranking Metric Embedding-based multicontextual behavior profiling method. The approach embeds user transaction behaviors into a ranking metric space to capture complex behavioral similarities. Compared to clustering-based anomaly detection, the model achieved improved precision and reduced false positive rates. While effective in contextual ranking, the framework's performance relies heavily on embedding quality, which can degrade under the rapid distribution shifts typical of modern fraud tactics.

Emerging research has also explored hybrid quantum-classical approaches. Wang et al. [4] integrated machine learning algorithms with quantum annealing solvers to improve fraud detection performance. Their Quantum-enhanced SVM demonstrated competitive accuracy and faster detection speed. Although promising, the approach depends on specialized quantum hardware accessibility, which currently limits its large-scale practical deployment in standard commercial banking infrastructures.

Deep learning architectures tailored for sequential data have also gained attention. Almazroi and Ayub [5] proposed a ResNeXt-embedded GRU (RXT-J) model for online payment fraud detection. The integration of convolutional feature extraction with gated recurrent units enabled superior temporal pattern learning. Nevertheless, the system mainly emphasizes temporal modeling and lacks the interpretability often required for fraud justification and regulatory compliance in financial environments.

Graph-based modeling has emerged as a powerful paradigm for relational analysis. Alarfaj and Shahzai [6] introduced a deep learning framework combining Graph Neural Networks (GNNs) with Autoencoders to capture dependencies among banking entities. While the model demonstrated improved precision and real-time detection efficiency, the complexity of maintaining a dynamic graph in high-velocity environments presents significant operational overhead compared to optimized tabular classifiers.

Expanding on this idea, Khosravi et al. [7] proposed an Attentional Spatial–Temporal GNN to jointly capture spatial relationships and temporal transaction evolution. By incorporating attention mechanisms, the model effectively identified coordinated fraud activities. Despite its strong performance, the intensive processing requirements for attentional mechanisms remain a practical hurdle for large-scale, low-latency financial applications compared to streamlined machine learning models.

Addressing the challenge of evolving tactics, Al Lawati et al. [8] proposed an integrated preprocessing and adaptive windowing framework using CNN combined with drift detection algorithms such as ADWIN and EDDM. This work emphasizes the importance of adaptive learning in dynamic environments. However, simpler models can often achieve similar results when combined with effective windowing techniques, making the high complexity of CNNs less necessary for many standard transaction datasets.

Privacy-preserving collaborative learning has also gained importance. Singh and Tripathy [9] introduced FedShield, a Federated Learning-based fraud detection framework for distributed systems. By enabling decentralized model training, the system achieved strong performance without sharing raw data. While federated learning enhances privacy, challenges related to communication overhead and model heterogeneity suggest that a centralized, optimized machine learning approach remains more stable for most initial college-level implementations.

Finally, synthetic dataset generation has been explored to address data scarcity. Baisholan et al. [10] introduced FraudX SimS, a synthetic dataset designed for anomaly detection in payment-card transactions. Using Random Forest as a baseline benchmark, the dataset demonstrated reliable evaluation metrics. This confirms that even when new data paradigms emerge, traditional machine learning remains the primary standard for evaluating fraud detection performance and reliability.

From the reviewed literature, it is evident that while significant progress has been made in fraud detection using advanced deep learning, graph-based methods, and distributed learning approaches, many systems focus on complexity-intensive techniques that may not be practical for standard implementations. There remains a need for a framework that combines high-precision, interpretable machine learning with structured preprocessing and efficient risk analysis. The proposed system aims to address this gap by adopting a machine learning approach using Logistic Regression and Random Forest for fraud transaction detection. By integrating input validation, feature engineering, train-test data splitting,

model training, fraud prediction, and risk evaluation, the system seeks to provide an accurate, scalable, and computationally efficient fraud detection solution. The inclusion of alert generation and dashboard visualization further supports practical financial transaction monitoring and real-time fraud prevention.

3.METHODOLOGY

For the proposed Fraud Transaction Detection system, an appropriate transactional dataset containing financial payment records was selected. The dataset consists of transaction attributes such as transaction amount, timestamp, merchant category, geographical indicators, and fraud labels, which are essential for identifying fraudulent patterns. This study utilizes structured transaction records to simulate a wide variety of fraud scenarios and ensure the models are tested against diverse transaction behaviors.

Financial fraud detection is inherently a highly imbalanced classification problem where fraudulent transactions represent only a small fraction of total transactions. Therefore, rigorous preprocessing techniques are applied to improve data quality and prepare the dataset for model training. Input validation, feature engineering, and correlation-based analysis are utilized to identify the most impactful transaction variables, improving detection performance and reducing bias toward legitimate transactions.

The methodology focuses on a comparative machine learning approach analyzing the performance of supervised learning algorithms including Logistic Regression and Random Forest. The proposed framework utilizes model training, fraud prediction, and risk evaluation to identify suspicious transactions with high efficiency. This approach combines structured preprocessing with predictive modeling to support accurate fraud detection and real-time transaction monitoring.

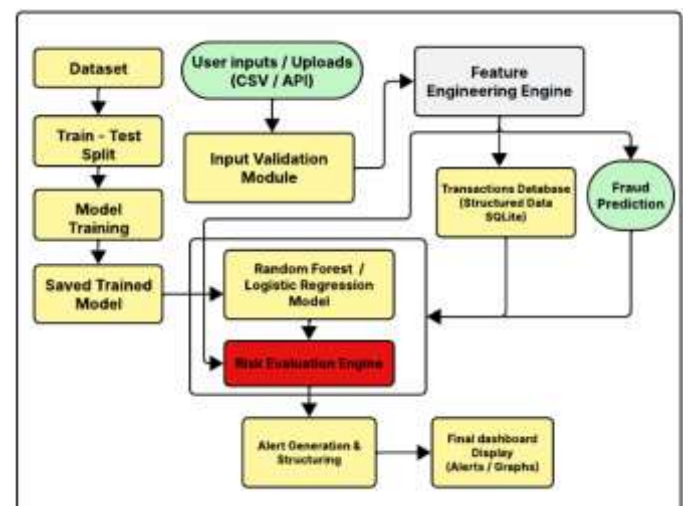


Fig. 1. Architecture Diagram

3.1 Understanding Stages of Machine Learning

The system begins by analyzing the collected transaction dataset to identify inconsistencies, missing values, and noise. Data preprocessing techniques such as normalization, encoding, and validation are applied before model training. After preprocessing, the dataset is divided into training and testing sets. Machine learning models are then trained to learn behavioral transaction patterns and classify transactions as fraudulent or legitimate. Model performance is evaluated using standard classification metrics to measure prediction effectiveness and robustness.

The steps followed are as:

Step 1: Data Collection

Step 2: Data Preprocessing and Visualization

Step 3: Feature Engineering

Step 4: Model Training and Creation

Step 5: Fraud Prediction and Risk Evaluation

Step 6: Alert Generation and Dashboard Monitoring Step 7: Performance Evaluation

3.2 Dependencies

The performance of the fraud detection framework depends heavily on the quality and diversity of transactional data used for training. Machine learning models require representative datasets that capture both legitimate and fraudulent behavioral patterns.

High-quality training data enables the model to learn hidden relationships among transaction attributes such as transaction amount, transaction timing, and account activity behavior. Balanced and informative datasets help improve fraud detection accuracy while minimizing false positives. Proper feature extraction and behavioral representation significantly enhance the system's capability to distinguish genuine transactions from fraudulent ones.

Machine learning algorithms become more effective when trained using relevant, consistent, and high-quality financial data, making data dependency one of the most critical aspects of the proposed system.

3.3 Web IDE

Google Colab is used as the primary development environment for implementing the proposed fraud detection system. Google Colab provides a cloud-based Jupyter notebook environment that allows developers to write and execute Python code directly through a web browser.

It offers computational resources that enable efficient model training without requiring high-performance local hardware. Colab also simplifies experimentation, visualization, and model evaluation.

3.4 Python Libraries

The proposed system utilizes several Python libraries for data processing, visualization, and machine learning implementation:

Pandas: A Python library used for data manipulation and analysis. It provides efficient tools for handling structured

datasets, cleaning data, and performing preprocessing operations.

NumPy: Used for numerical computations and efficient handling of large multidimensional arrays required for machine learning operations.

Seaborn: A statistical visualization library built on Matplotlib that helps in understanding data distributions, correlations, and fraud pattern visualization.

Matplotlib: A visualization library used to generate plots and graphical representations such as transaction trends, anomaly distributions, and performance metrics.

Scikit-learn (Sklearn): Provides machine learning algorithms for classification, preprocessing, model evaluation, and performance measurement.

3.5 Data Collection

Before training the machine learning models, the dataset is carefully examined to ensure data quality and consistency. Data preprocessing helps eliminate incorrect or noisy entries that may negatively affect model performance.

Dataset details:

Retrieved from the publicly available **Bank Transaction Dataset for Fraud Detection** available on Kaggle.

The dataset contains transaction attributes including timestamp, transaction amount, transaction type, account-related information, customer behavior indicators, and fraud labels.

The dataset is highly imbalanced, with fraudulent transactions forming a very small percentage of total records. Missing values are handled using preprocessing techniques such as imputation, normalization, and data cleaning.

Feature scaling and encoding methods are applied to prepare data for machine learning models.

	TransactionAmount	TransactionType	Location	Channel	CustomerAge	CustomerOccupation	TransactionDuration	LoginAttempts	AccountBalance	IsFraud
0	14.08	1	30	0	26	0	81	1	3112.01	0
1	216.26	1	70	0	44	0	181	1	12108.81	0
2	136.28	1	23	0	46	1	162	1	1132.20	0
3	164.50	1	33	0	29	0	20	1	8508.86	0
4	12.45	0	1	0	28	0	196	1	7424.40	0

Fig. 2. Dataset and its basic mathematical data.

3.6 Data Preprocessing

Data preprocessing plays a crucial role in improving the performance and reliability of fraud detection models. Financial transaction datasets often contain noisy, incomplete, and highly imbalanced data, which can negatively affect machine learning performance if not properly handled. Therefore, several preprocessing techniques are applied before model training.

Data Cleaning: Data cleaning involves removing incorrect, inconsistent, duplicate, or corrupted transaction records from the dataset. Financial transaction data may contain redundant entries or formatting inconsistencies that can affect model performance. Cleaning ensures that transaction attributes such as timestamps, transaction amounts, and account-related fields are accurate and structured properly for analysis.

Missing Values: Missing values may occur due to incomplete transaction logs or data collection issues. Such values are handled using appropriate preprocessing methods such as removal of incomplete records or simple imputation where necessary. Proper handling of missing data prevents bias and improves model stability.

Outlier Detection and Analysis: Outliers in financial data may represent either suspicious behavior or data entry errors. Statistical analysis techniques are used to identify abnormal values in transaction-related attributes. Fraud-related anomalies are preserved for model learning, while erroneous outliers caused by recording mistakes are handled to improve data consistency.

Feature Encoding: Transaction datasets often include categorical variables such as transaction type or merchant category. Encoding techniques are applied to convert categorical features into numerical form so that they can be processed effectively by machine learning algorithms.

Normalization: Transaction features such as payment amount, account balances, and transaction frequency often vary widely in scale. Normalization techniques are applied to transform features into a common range, improving model convergence and classification performance for algorithms sensitive to feature magnitude.

Correlation Analysis: Correlation analysis is used to identify relationships among transactional attributes and determine the most influential features for fraud prediction. This helps reduce redundancy and improves model efficiency by focusing on significant variables during feature engineering.

Data Visualization: Data visualization helps in understanding transaction behavior, fraud distribution, and anomaly patterns within the dataset. Visualization techniques highlight correlations, trends, and suspicious activity indicators. Libraries such as Matplotlib and Seaborn are used to generate graphs including transaction distributions, correlation heatmaps, and fraud frequency plots, enabling better exploratory data analysis and model interpretation.

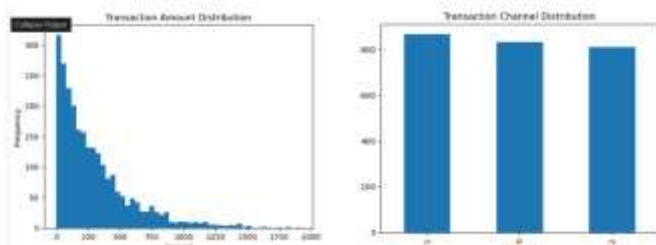


Fig. 3. Transaction Amount Distribution Analysis During Data Preprocessing

3.7 Splitting Data

The train-test split technique is used to evaluate machine learning models designed for **fraud prediction**. This method allows the performance of the developed model to be validated using unseen transaction data, ensuring that the model generalizes well to real-world financial environments. The processed dataset is divided into two subsets: a training set and a testing set. The training dataset is used to learn behavioral transaction patterns, while the testing dataset is used to evaluate prediction accuracy and model robustness.

In the proposed system, the dataset is split in an **80:20** ratio, where 80% of the data is used for training and 20% is reserved for testing. Since fraud detection datasets are often imbalanced, **stratified sampling** is applied to maintain the same proportion of fraudulent and legitimate transactions in both datasets, ensuring balanced model evaluation.

The splitting process is implemented using the `train_test_split()` function from the **Scikit-learn model_selection** module, ensuring reproducibility and

consistent evaluation. The dataset contains structured transaction attributes such as transaction amount, transaction type, timestamps, customer-related indicators, and fraud labels, making it suitable for supervised machine learning classification.

3.8 Machine Learning Model

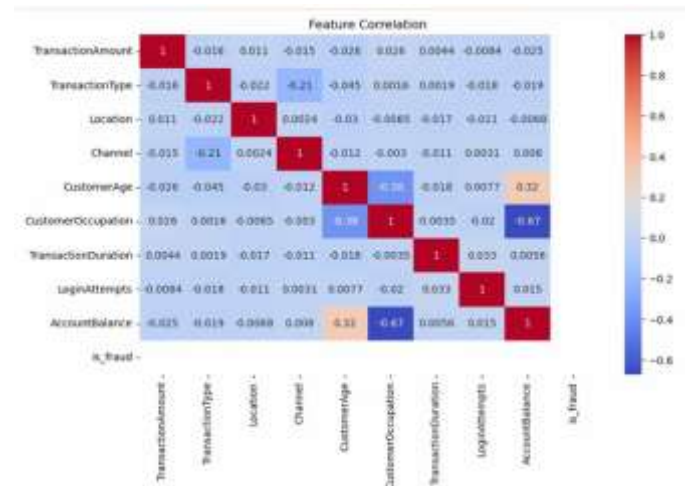


Fig. 4. Correlation Matrix

From the above Feature Correlation Matrix, relationships among different transactional attributes used for fraud detection can be observed. The heatmap illustrates how strongly features such as transaction amount, account balances, and balance changes are related to each other and to the fraud indicator variable.

A strong positive correlation can be observed between balance-related attributes, indicating that account balances before and after transactions are naturally dependent variables. Similarly, transaction amount shows significant correlation with balance change features, suggesting that transaction value plays an important role in determining financial behavior patterns.

The correlation matrix also highlights that fraud-related attributes exhibit distinct relationships compared to normal transactional features. Fraudulent transactions tend to disrupt normal balance transitions, resulting in irregular correlations between amount-related features and account balances. These abnormal dependencies provide valuable indicators for identifying suspicious activities.

Certain features demonstrate weaker correlations, indicating that they may contribute less directly to fraud prediction. Feature selection based on correlation analysis helps reduce redundancy and improves model efficiency by focusing on influential attributes. Understanding these relationships enables the models to capture behavioral inconsistencies and transactional anomalies more effectively.

Logistic Regression: Logistic Regression is a supervised learning algorithm widely used for binary classification problems such as fraud detection. It estimates the probability of a transaction being fraudulent or legitimate using a logistic function. The model learns relationships between transaction features and fraud labels by minimizing classification error and serves as an effective baseline predictive model due to its simplicity, computational efficiency, and interpretability.

Random Forest Classifier: Random Forest is an ensemble learning technique that combines multiple decision trees to improve prediction accuracy and robustness. Each tree is trained on different subsets of data, and the final prediction is determined through majority voting. This approach reduces overfitting and enhances fraud detection performance by capturing complex behavioral patterns across transactions.

The proposed system utilizes both Logistic Regression and Random Forest for comparative analysis to identify the model that provides better fraud detection performance. These models are trained using processed transaction data and evaluated using classification metrics to determine their effectiveness in identifying fraudulent transactions while minimizing false positives.

3.9 Model Evaluation

The performance of the proposed fraud detection system is evaluated using standard classification metrics that measure the effectiveness of the machine learning models in identifying fraudulent transactions while minimizing misclassification.

Accuracy: Accuracy measures the proportion of correctly classified transactions out of the total number of transactions processed. It indicates the overall prediction performance of the model by comparing correct fraud and non-fraud predictions against all predictions made.

Precision: Precision measures the proportion of transactions predicted as fraudulent that are actually fraudulent. It is an important metric in fraud detection because high precision helps reduce false positives, preventing legitimate transactions from being incorrectly flagged as suspicious.

Recall: Recall measures the proportion of actual fraudulent transactions that are correctly identified by the model. This metric is critical in fraud detection because a higher recall ensures that more fraudulent transactions are detected, reducing the risk of financial losses due to missed fraud cases. **F1-Score:** F1-score is the harmonic mean of Precision and Recall, providing a balanced measure of model performance. It is especially useful for highly imbalanced datasets such as fraud detection, where both minimizing false positives and maximizing fraud detection are equally important.

These evaluation metrics are used to compare the performance of Logistic Regression and Random Forest models and determine the most effective model for accurate and reliable fraud transaction detection.

3.10 Results

Table 2: Performance Comparison of Machine Learning Models

Model	AUC-ROC (%)	Precision (%)	Recall (%)	F1-Score (%)	Average Precision (%)
Logistic Regression	90.9	47.4	75.0	58.1	73.7
Random Forest	95.8	82.8	80.0	81.4	85.0

The experimental results demonstrated that the proposed fraud detection framework is effective in identifying fraudulent transactions using machine learning techniques. A comparative evaluation was performed using Logistic Regression and Random Forest to determine their performance across standard classification metrics.

The results indicate that Random Forest outperformed Logistic Regression across all major evaluation metrics, providing stronger fraud detection capability and better classification performance. Logistic Regression served as a reliable baseline model, while Random Forest demonstrated superior performance in capturing complex transactional fraud patterns. The results show that Random Forest achieved the highest overall performance, with an AUC-ROC of 95.8% and F1-score of 81.4%, indicating stronger classification capability compared to Logistic Regression. Higher precision and average precision values further indicate improved fraud identification with fewer false positives. Feature analysis also shows that transaction-related attributes and engineered features significantly contribute to model performance. The integration of preprocessing, feature engineering, risk evaluation, and alert generation enhances the reliability of the proposed fraud detection framework and supports practical financial transaction monitoring.

4. CONCLUSION

Based on experimental evaluation, the proposed fraud transaction detection framework demonstrates effective performance in identifying fraudulent transactions using machine learning techniques. The system utilizes Logistic Regression and Random Forest models combined with data preprocessing, feature engineering, risk evaluation, and alert generation to provide a structured and scalable fraud detection solution for financial transaction monitoring. Among the implemented models, Random Forest achieved the best overall performance with an AUC-ROC of 95.8%, F1-score of 81.4%, Precision of 82.8%, Recall of 80.0%, and Average Precision of 85.0%, outperforming Logistic Regression across all major evaluation metrics. Logistic Regression achieved an AUC-ROC of 90.9%, F1-score of 58.1%, Precision of 47.4%, Recall of 75.0%, and Average Precision of 73.7%, demonstrating reliable baseline classification performance with computational efficiency and interpretability.

The comparative analysis shows that Random Forest provides improved fraud detection capability by capturing complex transactional behavior patterns, while Logistic Regression serves as a strong baseline model for binary fraud classification. The integration of preprocessing, correlation-based feature engineering, and structured model evaluation improves detection reliability while reducing false positives and missed fraud cases. The risk evaluation engine and alert generation module further strengthen the practical applicability of the system for monitoring suspicious transactions in near real-time environments. Overall, the proposed framework provides an accurate, scalable, and computationally efficient fraud detection solution suitable for modern financial applications. Future work may

focus on real-time deployment, explainable machine learning techniques, and further optimization using larger and more diverse financial transaction datasets.

REFERENCES

- [1]N. Upadhyay *et al.*, "Machine Learning Perspective: Fraud Payment Transaction Detection," in *Journal of Mobile Multimedia*, vol. 21, no. 3-4, pp. 577-598, July 2025, doi: 10.13052/jmm1550-4646.213414.
- [2]Can Liu, Qiwei Zhong, Xiang Ao, Li Sun, Wangli Lin, Jinghua Feng, Qing He, and Jiayu Tang. 2020. Fraud Transactions Detection via Behavior Tree with Local Intention Calibration. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '20). Association for Computing Machinery, New York, NY, USA, 3035–3043. <https://doi.org/10.1145/3394486.3403354>.
- [3]J. Cui, C. Yan and C. Wang, "ReMEMBeR: Ranking Metric Embedding-Based Multicontextual Behavior Profiling for Online Banking Fraud Detection," in *IEEE Transactions on Computational Social Systems*, vol. 8, no. 3, pp. 643-654, June 2021, doi: 10.1109/TCSS.2021.3052950.
- [4.]H. Wang, W. Wang, Y. Liu and B. Alidaee, "Integrating Machine Learning Algorithms With Quantum Annealing Solvers for Online Fraud Detection," in *IEEE Access*, vol. 10, pp. 75908-75917, 2022, doi: 10.1109/ACCESS.2022.3190897.
- [5]A. A. Almazroi and N. Ayub, "Online Payment Fraud Detection Model Using Machine Learning Techniques," in *IEEE Access*, vol. 11, pp. 137188-137203, 2023, doi: 10.1109/ACCESS.2023.3339226
- [6]F. Khaled Alarfaj and S. Shahzadi, "Enhancing Fraud Detection in Banking With Deep Learning: Graph Neural Networks and Autoencoders for Real-Time Credit Card Fraud Prevention," in *IEEE Access*, vol. 13, pp. 20633-20646, 2025, doi: 10.1109/ACCESS.2024.3466288.
- [7]Khosravi, S., Kargari, M., Teimourpour, B. *et al.* Transaction fraud detection via attentional spatial-temporal GNN. *J Supercomput* 81, 537 (2025). <https://doi.org/10.1007/s11227-025-06983-8>
- [8]H. M. R. A. Lawati *et al.*, "An Integrated Preprocessing and Drift Detection Approach With Adaptive Windowing for Fraud Detection in Payment Systems," in *IEEE Access*, vol. 13, pp. 92036-92056, 2025, doi: 10.1109/ACCESS.2025.3569609.
- [9]Singh, N., Tripathy, S. *FedShield*: federated learning based robust online payment fraud detection. *J Supercomput* 81, 1289 (2025). <https://doi.org/10.1007/s11227-025-07774-x>
- [10]N. Baisholan, J. Eric Dietz, S. Gnatyuk, M. Turdalyuly, E. T. Matson and K. Baisholanova, "FraudX SimS: A Synthetic Dataset for Anomaly Detection in Payment-Card Transactions," in *IEEE Access*, vol. 13, pp. 208549-208562, 2025, doi: 10.1109/ACCESS.2025.3637828.