International Journal of Science and Technology (IJST) | ISSN(O): 2049-7318 |2(4):39-44, 2026 | DOI:10.5281/zenodo.21404651

Transaction Fraud detection using machine learning with explainable AI

K Jeevan Babu, G Sujith kumar, V Sohel, O Nithin

Department of Computer Science and Engineering
Under the Guidance of Mr, B Madhusudan Reddy
Aditya College of Engineering, Madanapal

Abstract: Financial transaction fraud has emerged as a major concern in modern digital payment ecosystems due to the rapid growth of online banking, mobile payments, and e-commerce platforms. As the volume of transactions increases, so does the complexity and frequency of fraudulent activities, leading to significant financial losses and reduced customer trust. Traditional rule-based fraud detection systems are limited in their ability to identify new and evolving fraud patterns, making them inefficient in dynamic real-world environments. To address these challenges, this project proposes a machine learning-based approach for detecting fraudulent transactions using the PaySim dataset. The system is designed as a complete end-to-end pipeline that includes data preprocessing, exploratory data analysis, feature engineering, and handling of class imbalance using techniques such as SMOTE. Multiple supervised learning algorithms, including Logistic Regression, Random Forest, XGBoost, and LightGBM, are implemented and evaluated to identify the most effective model for fraud detection. In addition to achieving high predictive performance, the project emphasizes interpretability through the integration of Explainable Artificial Intelligence (XAI) techniques. SHAP (SHapley Additive Explanations) is used to provide clear insights into model predictions, helping users understand why specific transactions are classified as fraudulent. Furthermore, an interactive Streamlit dashboard is developed to visualize results and enable real-time testing. The overall system offers a scalable, accurate, and transparent solution for real-world financial fraud detection applications.

INTRODUCTION

Financial transaction fraud has emerged as a major concern in modern digital payment ecosystems due to the rapid growth of online banking, mobile payments, and e-commerce platforms. As the volume of transactions increases, so does the complexity and frequency of fraudulent activities, leading to significant financial losses and reduced customer trust. Traditional rule-based fraud detection systems are limited in their ability to identify new and evolving fraud patterns, making them inefficient in dynamic real-world environments. To address these challenges, this project proposes a machine learning-based approach for detecting fraudulent transactions using the PaySim dataset. The system is designed as a complete end-to-end pipeline that includes data preprocessing, exploratory data analysis, feature engineering, and handling of class imbalance using techniques such as SMOTE. Multiple supervised learning algorithms, including Logistic Regression, Random Forest, XGBoost, and LightGBM, are implemented and evaluated to identify the most effective model for fraud detection. In addition to achieving high predictive performance, the project emphasizes interpretability through the integration of Explainable Artificial Intelligence (XAI) techniques. SHAP (SHapley Additive

Explanations) is used to provide clear insights into model predictions, helping users understand why specific transactions are classified as fraudulent. Furthermore, an interactive Streamlit dashboard is developed to visualize results and enable real-time testing. The overall system offers a scalable, accurate, and transparent solution for real-world financial fraud detection applications.

The foundation of this project lies in machine learning techniques used for detecting fraudulent financial transactions. Supervised learning algorithms such as Logistic Regression, Random Forest, XGBoost, and LightGBM are utilized to classify transactions as either legitimate or fraudulent. These models are capable of learning complex patterns from historical transaction data and adapting to evolving fraud behaviors. Among these, boosting algorithms like XGBoost imbalanced, where fraudulent transactions represent only a small portion of the data. To address this issue, SMOTE (Synthetic Minority Over-sampling Technique) is applied to generate synthetic samples of the minority class, improving the model's ability to detect fraud. Additionally, feature engineering techniques are used to create meaningful variables from raw data, enhancing model performance and accuracy.

Explainable Artificial Intelligence (XAI) plays a crucial role in making the system transparent and trustworthy. SHAP (SHapley Additive

Explanations) is used to interpret the predictions of machine learning models by showing how each feature contributes to the final decision. Along with this, a Streamlit-based web application is developed to provide an interactive interface for users to visualize results and test transactions in real time. These technologies together ensure that the system is not only accurate but also understandable and user-friendly.

With the rapid advancement of digital technologies, financial transactions have increasingly shifted toward

online platforms such as mobile banking, digital wallets, and e commerce systems. While this transformation has improved convenience and accessibility, it has also created new opportunities for fraudulent activities. Fraudsters continuously exploit system vulnerabilities using sophisticated techniques, making it difficult for traditional security mechanisms to keep up with evolving threats.

RELATED WORK

2.1 Rule- Based Fraud detection Systems

Traditional fraud detection systems rely on predefined rules such as transaction limits, unusual locations, or frequency thresholds to identify suspicious activities. These systems are simple to implement and easy to understand. However, they lack adaptability and fail to detect new and evolving fraud patterns, resulting in high false positives and missed fraud cases.

2.2 Traditional Machine Learning Model

Machine learning techniques such as Logistic Regression, Decision Trees, and Random Forest have been widely used for fraud detection.

These models learn patterns from historical transaction data and provide better flexibility than rule-based systems. However, they often struggle with highly imbalanced datasets and may fail to capture complex fraud patterns effectively.

2.3 Static Analytical Systems

Static analytical approaches use statistical methods and fixed models to detect anomalies in transaction data. These systems analyze historical trends and flag deviations as potential fraud. While they are computationally simple, they lack the ability to adapt to changing fraud strategies and become less effective over time.

2.4 Ensemble and Boosting-Based Models

Advanced machine learning models such as Random Forest, XGBoost, and LightGBM combine multiple weak learners to improve prediction accuracy. These models are capable of handling large datasets and capturing complex relationships between features. However, they require careful tuning and higher computational resources.

2.5 Imbalance Handling Techniques in Fraud Detection

Smart contracts have been used to automate file- sharing permissions, ensuring that access is granted only when predefined conditions are met. These systems reduce manual intervention and improve trust. However, poorly designed smart contracts may introduce vulnerabilities.

2.6 Data Integrity Verification Techniques Hashing algorithms like SHA-256 are commonly used to verify file integrity. Any modification in the file results in a change in the hash value, ensuring tamper detection. However, without decentralized storage, integrity verification alone cannot prevent data loss or unauthorized deletion.

2.7 Limitations of Existing Systems

Despite advancements, existing solutions face challenges such as:

- Inability to adapt to evolving fraud patterns
- High false positive and false negative rates
- Difficulty in handling imbalanced datasets
- Lack of transparency in complex models
- Scalability issues with large transaction data

LITERATURE REVIEW

3.1 Evolution of Fraud Detection Systems

Fraud detection systems have undergone In the early stages, fraud detection was primarily performed through manual inspection and basic rule-based mechanisms. Financial institutions relied heavily on human auditors and predefined rules such as transaction limits, unusual spending behavior, and location-based checks to identify suspicious activities. While these methods were effective for detecting straightforward fraud cases, they were timeconsuming, labor-intensive, and prone to human error. Moreover, as the volume of financial transactions increased with the growth of digital banking and online payment systems, these traditional approaches became inefficient and difficult to scale.

To overcome the limitations of manual and rulebased systems, statistical and analytical methods were introduced. These systems used historical transaction data to identify patterns and anomalies through techniques such as threshold analysis and basic probability models. Although these approaches improved detection efficiency, they were still limited in their ability to adapt to new and evolving fraud strategies. Fraudsters continuously modify their techniques to bypass existing detection rules, making static systems less reliable over time.

3.2 Machine Learning Techniques in Fraud Detection

Machine learning has emerged as one of the most powerful and widely adopted technologies for detecting fraudulent transactions in modern financial systems. With the rapid growth of digital payments, online banking, and ecommerce platforms, the volume of transaction data has increased exponentially. Traditional rule-based systems are no longer sufficient to handle this complexity, as they fail to adapt to evolving fraud patterns. Machine learning addresses these challenges by automatically analyzing large volumes of data, identifying hidden patterns, and making accurate predictions without relying on predefined rules. Supervised learning algorithms play a crucial role in fraud detection, as they are trained using labeled datasets where transactions are classified as either fraudulent or legitimate. Models such as Logistic Regression,

Decision Trees, Random Forest, XGBoost, and LightGBM are commonly used for classification tasks.

Logistic Regression provides a simple and interpretable baseline model, while Decision Trees and Random Forest offer better performance by capturing non-linear relationships. Ensemble methods like Random Forest combine multiple decision trees to reduce overfitting and improve generalization. Boosting algorithms such as XGBoost and LightGBM further enhance performance by sequentially learning from previous errors, making them highly effective in detecting complex fraud patterns.

3.3 Handling Imbalanced Datasets

One of the most critical challenges in fraud detection systems is the issue of class imbalance in datasets. In real-world financial transaction data, fraudulent transactions typically represent only a very small percentage of the total transactions, often less than 1%, while the vast majority of transactions are legitimate. This highly skewed distribution creates a significant problem for machine learning models, as they tend to become biased toward the majority class (non-fraud) and may fail to correctly identify rare but important fraudulent cases.

When trained on imbalanced datasets, many machine learning models prioritize overall accuracy rather than correctly identifying minority class instances. As a result, a model may achieve very high accuracy by simply predicting all transactions as legitimate, while completely missing fraudulent transactions. This leads to poor recall for the fraud class and reduces the effectiveness of the system in realworld applications, where detecting fraud is far more critical than correctly classifying normal transactions.

To address this issue, various data-level and algorithm-level techniques have been developed. One of the most widely used approaches is SMOTE (Synthetic Minority Over-sampling Technique). SMOTE works by generating synthetic samples of the minority class (fraudulent transactions) based on existing data points. Instead of simply duplicating minority samples, SMOTE creates new synthetic instances by interpolating between nearby minority class examples.

Another important approach is the use of algorithm-level methods such as class weighting and cost-sensitive learning. In these methods, higher importance or penalty is assigned to misclassification of fraudulent transactions, forcing the model to pay more attention to the minority class. This improves the model's ability to detect fraud without significantly affecting overall performance.

3.4 Performance Evaluation Metrics

Evaluating the performance of fraud detection models is a critical step in building an effective and reliable system. In fraud detection, relying solely on accuracy as a performance metric can be highly misleading, especially due to the presence of imbalanced datasets where fraudulent transactions constitute only a small fraction of the total data. A model can achieve very high accuracy by simply predicting all transactions as legitimate, yet completely fail to identify actual fraud cases. Therefore, more comprehensive evaluation metrics are required to properly assess the effectiveness of machine learning models in detecting fraud. One of the most important metrics used in fraud detection is Precision, which measures the proportion of correctly identified fraudulent transactions out of all transactions predicted as fraud. High precision indicates that the model makes fewer false positive errors, meaning legitimate transactions are not incorrectly flagged as fraudulent. This is important in reducing unnecessary alerts and maintaining customer satisfaction.

Another crucial metric is Recall (also known as sensitivity or true positive rate), which measures the proportion of actual fraudulent transactions that are correctly identified by the model. Recall is particularly important in fraud detection because missing a fraudulent transaction can lead to significant financial losses.

4. PROBLEM STATEMENT

In recent years, the rapid growth of digital payment systems, online banking, and ecommerce platforms has significantly increased the volume of financial transactions. While these advancements have improved convenience and accessibility, they have also led to a rise in fraudulent activities.

Transaction fraud has become a major challenge for financial institutions, resulting in substantial financial losses, reduced customer trust, and increased operational costs. According to the project context, fraud detection is critical to ensure secure and reliable financial systems. Traditional fraud detection methods primarily rely on rule-based systems and manual verification processes. These systems use predefined rules such as transaction limits, unusual patterns, or location-based checks to identify suspicious activities. However, such approaches are rigid and cannot adapt to new and evolving fraud techniques. As fraudsters continuously modify their strategies, rule-based systems become less effective over time and often fail to detect complex and hidden fraud patterns. Additionally, manual auditing is timeconsuming, expensive, and not scalable for handling large volumes of transactions. Another major challenge in fraud detection is the issue of class imbalance, where fraudulent transactions represent only a small fraction of the overall dataset. This imbalance leads to biased machine learning models that tend to favor the majority class (legitimate transactions), resulting in poor detection of actual fraud cases.

5. PROPOSED SYSTEM

The proposed system aims to develop an intelligent and automated solution for detecting fraudulent financial transactions using advanced machine learning techniques. Unlike traditional rule-based systems, this approach leverages data-driven models to identify complex and hidden fraud patterns with higher accuracy and efficiency. The system is designed as a complete end-to-end pipeline that processes raw transaction data, extracts meaningful insights, and classifies transactions as either fraudulent or legitimate in real time.

The system begins with the data collection and preprocessing stage, where transaction data is gathered from datasets such as PaySim. The collected data is cleaned by handling missing values, removing inconsistencies, and transforming it into a structured format suitable for analysis. Categorical variables are encoded into numerical values, and feature scaling is applied to normalize the data. This preprocessing step ensures that the dataset is accurate, consistent, and ready for further analysis and model training.

Next, the system performs Exploratory Data Analysis (EDA) and feature engineering to understand patterns and relationships within the data. Important features such as transaction amount, type, and account balance changes are analyzed to identify potential fraud indicators. Feature engineering helps in creating meaningful attributes that improve the performance of machine learning models. This stage plays a crucial role in enhancing the accuracy and robustness of the system. To address the challenge of imbalanced datasets, the proposed system incorporates the SMOTE (Synthetic Minority Over-sampling Technique). This technique generates synthetic samples of fraudulent transactions, balancing the dataset and enabling the models to learn fraud patterns more effectively. By handling class imbalance, the system significantly improves its ability to detect rare and critical fraud cases.

6. SYSTEM ARCHITECTURE

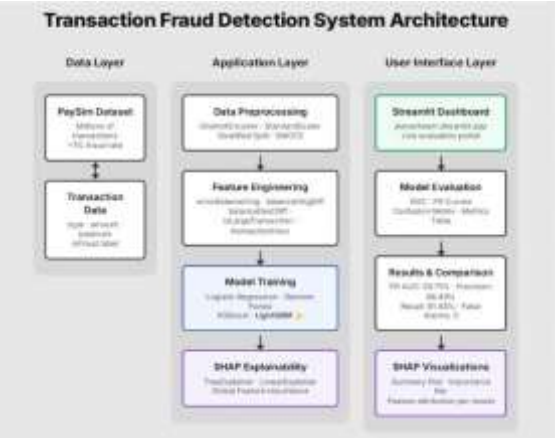


Fig 6.1:System Architecture

The system architecture of the proposed transaction fraud detection system is designed using a layered approach to ensure efficient data processing and accurate prediction. It consists of three main layers: the Data Layer, Application Layer, and User Interface Layer. Each layer performs a specific function and works together to provide a complete end-to-end fraud detection solution. This structured design improves scalability, maintainability, and performance of the system.

The Data Layer is responsible for managing and storing the transaction dataset used in the system. It includes the PaySim dataset, which contains a large number of transaction records with details such as transaction type, amount, account balances, and fraud labels. This layer ensures that the data is properly organized and available for further processing. It acts as the foundation of the system by providing reliable and structured input data to the application layer.

The Application Layer performs the core operations of the system, starting with data preprocessing and feature engineering. In the preprocessing stage, the raw data is cleaned, transformed, and prepared using techniques such as encoding and scaling. The dataset is also split into training and testing sets, and SMOTE is applied to handle class imbalance. Feature engineering is then used to create meaningful attributes like balance differences and transaction patterns, which improve the model's ability to detect fraud.

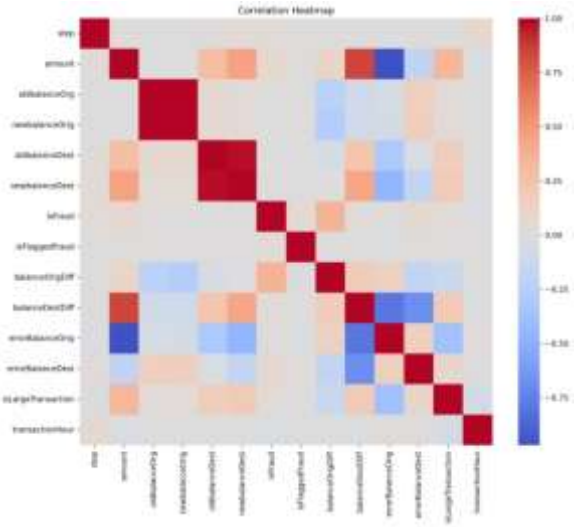
The Application Layer performs the core operations of the system, starting with data preprocessing and feature engineering. In the preprocessing stage, the raw data is cleaned, transformed, and prepared using techniques such as encoding and scaling. The dataset is also split into training and testing sets, and SMOTE is applied to handle class imbalance. Feature engineering is then used to create meaningful attributes like balance differences and transaction patterns, which improve the model's ability to detect fraud.

The User Interface Layer provides an interactive platform for users through a Streamlit dashboard. Users can input transaction details or upload datasets and receive real-time fraud predictions. This layer also displays performance evaluation metrics such as precision, recall, ROC curves, and confusion matrix, along with SHAP visualizations. Overall, the architecture ensures smooth data flow, accurate predictions, and a user-friendly experience suitable for real-world fraud detection applications.

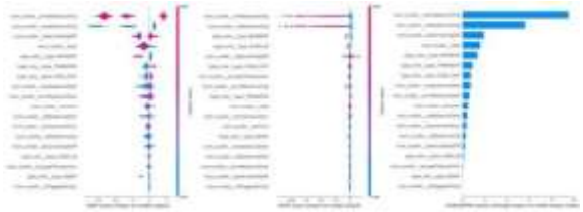
7. Results

SS 7.1: User Login

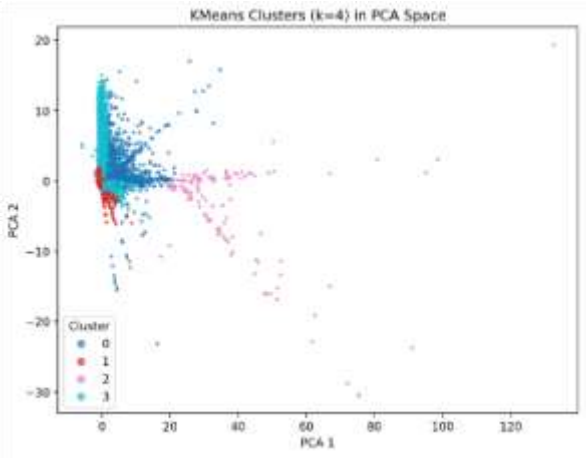
SS 7.2: Main Dashboard



SS 7.3: Heat Map



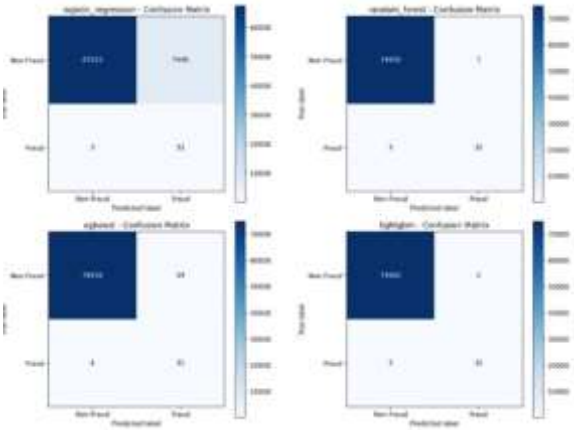
SS 7.4: SHAP Value



SS 7.5: K-Means Clustering with PCA

SS 7.6:Upload File Page

SS 7.7: Model Comparasion



SS 7.8: Confusion Matrix

8. APPLICATIONS

8.1 Banking and Financial Systems

The proposed system can be used in banks and financial institutions to detect fraudulent transactions in real time. It helps analyze large volumes of transaction data and accurately identify suspicious activities. By using machine learning models, the system reduces financial losses and improves security in digital payment systems.

8.2 Online Payment Platforms

The system can be integrated into online payment gateways and mobile wallets to monitor transactions continuously. It ensures that fraudulent activities such as unauthorized transfers, unusual spending patterns, and fake transactions are detected instantly, enhancing user trust and platform reliability.

8.3 E-commerce Systems

The system enables secure management and sharing of academic records such as certificates, transcripts, and

research data. It prevents forgery and ensures that only verified users can access or modify the data.

8.4 Insurance Sector

The system can be applied in insurance companies to detect fraudulent claims by analyzing transaction and claim-related data. It helps in identifying suspicious patterns and reduces false claims, improving efficiency and saving costs for insurance providers.

8.5 Government and Tax Systems

Government organizations can use this system to monitor financial transactions and detect fraud in taxation, subsidies, and public fund usage. It ensures transparency and helps in preventing misuse of public resources.

8.6 Corporate and Enterprise Systems Organizations can use the system to monitor internal financial transactions and detect anomalies. It helps in preventing insider fraud, unauthorized transactions, and financial mismanagement, ensuring secure and transparent operations.

8.7 Real-Time Monitoring Systems The system supports real-time fraud detection, allowing organizations to take immediate action such as blocking transactions or flagging suspicious activities.

REFERENCES

- S. Bhattacharyya, S. Jha, K. Tharakunnel, and J. C. Westland, "Data Mining for Credit Card Fraud: A Comparative Study," *Decision Support Systems*, vol. 50, no. 3, pp. 602–613, 2011.
- Abdallah, M. A. Maarof, and A. Zainal, "Fraud Detection System: A Survey," *Journal of Network and Computer Applications*, vol. 68, pp. 90–113, 2016.
- P. Phua, V. Lee, K. Smith, and R. Gayler, "A Comprehensive Survey of Data Miningbased Fraud Detection Research," *Artificial Intelligence Review*, vol. 34, no. 1, pp. 1–14, 2010.
- C. Whitrow, D. J. Hand, P. Juszczak, D. Weston, and N. M. Adams, "Transaction Aggregation as a Strategy for Credit Card Fraud Detection," *Data Mining and Knowledge Discovery*, vol. 18, no. 1, pp. 30– 55, 2009.
- Roy et al., "Credit Card Fraud Detection Using Machine Learning and Deep Learning," *IEEE Conference*, 2021.
- Abdallah et al., "Fraud Detection Using Machine Learning: A Systematic Review," *IEEE Access*, 2021.
- Ali et al., "Explainable Artificial Intelligence for Fraud Detection in Banking Transactions," *IEEE Access*, 2022.
- T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," *Proceedings of the 22nd ACM SIGKDD Conference*, 2016.