2025 International Conference on Computational Robotics, Testing and Engineering Evaluation (ICCRTEE) | 979-8-3315-9709-2/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICCRTEE64519.2025.11052942

Comparative Analysis of UPI Fraud Detection Using Ensemble Learning

Dontha Madhusudhana Rao
Department of Computer Science and Engineering
Amrita Vishwa Vidyapeetham
Amaravati, India
madhu.dontha@gmail.com

Talluri Ranga Sai Varun
Department of Computer Science and Engineering
Amrita Vishwa Vidyapeetham
Amaravati, India
trangasaivarun@gmail.com

Telagamsetty Viswajith Gupta
Department of Computer Science and Engineering
Amrita Vishwa Vidyapeetham
Amaravati, India
viswajithguptat17@gmail.com

Dokala Manoj Kumar
Department of Computer Science and Engineering
Amrita Vishwa Vidyapeetham
Amaravati, India
manojdokala215@gmail.com

***Abstract*—The rapid rise of the Unified Payments Interface has made digital transactions faster and more convenient, but it has also led to a significant increase in fraudulent activities. This study explores two separate transactional data sets with extensive transactional data, followed by intensive preprocessing steps with advanced feature engineering and class balancing to fix internal quality concerns of data. A thorough evaluation of Machine Learning models is done including traditional algorithms like decision tree, Logistic Regression, and Naive Bayes, advanced Ensemble methods such as Random Forest, Gradient Boosting, AdaBoost, Bagging, Extra Trees, XGBoost, and CatBoost. Performance is assessed using metrics such as precision, recall, F1 score, ROC-AUC, specificity, and accuracy. Ensemble methods like XGBoost and CatBoost, frequently achieved above 99% accuracy with very few misclassifications. The results indicate that Ensemble-based approaches could effectively detect fraudulent activities and, most importantly, reduce financial losses, build confidence within customers, and pave the way for managing risk proactively with financial institutions and digital payment platforms.**

Keywords—UPI, XGBoost, Imbalanced Dataset, Ensemble Learning, SMOTE, Hold-Out, K-fold cross-validation.

I. Introduction

Unified Payments Interface (UPI) is a real-time digital payment system developed under the patronage of RBI by NPCI [1]. It was launched on 11 April 2016, by then-RBI Governor Dr. Raghuram Rajan, with banks presenting the UPI-enabled apps on 25 August 2016 [2]. UPI allows transferring money, paying merchants, or transacting peer-to-peer, linking multiple bank accounts to a single mobile application. It works through IMPS and AEPS to effect instant settlements and boosts digital commerce by optimizing cash flow and customer experience [3]. Fraudulent activities in UPI have also increased along with the increase in its popularity and usage.

In a Parliament report, the Ministry of Finance highlighted the growing risks. In the first months of FY25, 6.32 lakh cases of fraud with Rs. 485 crores already stand reported [4].

Before the advent of Machine Learning (ML), rule-based systems, manual analyses, and crude statistical analyses worked their way up to UPI fraud detection. They flagged transactions based on rigid criteria. But were severely limited in their ability to catch sophisticated fraud because of their inflexible threshold values [5]. Fraud detection faces challenges because these systems rely on historical data, and as a result, they often misidentify real transactions as fraudulent and fail to catch new fraud techniques. The traditional and AI methods can complement each other in acquiring enhanced, efficient, and flexible detection of fraud [6].

ML helps secure UPI by recognizing suspicious patterns, preventing real-time fraud, and assigning risk scores low enough to ensure minimal false positives. Thus, ML keeps changing with new tactics of fraud, while behavior analysis provides an additional layer of security against unauthorized access for safer digital transactions [7]. There are some hurdles that ML meets while trying to detect fraud. The task may require a large dataset for its efforts. While modeling, there could be an introduction of biases as well as poor transparency of such models [8].

Through the process of combining different models, Ensemble Learning (EL) enhances UPI fraud detection by increasing the accuracy of the model, preventing it from being overfitting and obtaining robustness against imbalanced and noisy data. In this paradigm, the identification of frauds is enhanced, as is the stability of a model [9].

The purpose of this study is to develop an efficient and high-performance system for detecting fraud with high accuracy while minimizing false positives and identifying fraudulent patterns. The area was concerned with assessing the fraud detection system in digital transactions with the XGBoost algorithm. The model analyzes transaction frequency and amount deviations after, a thorough data preprocessing, including feature engineering, encoding, scaling, data balancing, and training on a large dataset, so that it efficiently detects anomalies. The application features a user-friendly interface for inputting transaction data and verifying authenticity. The solution was developed to improve the UPI ecosystem against future digital payment attacks, and further steps for the implementation of the model are presented.

979-8-3315-9709-2/25/$31.00 ©2025 IEEE

II. Literature Survey

This section briefly outlines recent research in the field of UPI fraud detection suggesting that a large portion of the previous work was mainly based on sophisticated Machine and Deep Learning (DL) models to catch intricate fraud patterns.

S. K. Lokesh Naik et al. [7] have studied the role of AI and ML in improving the security of UPI transactions by proposing a real-time fraud detection system based on Convolutional Neural Networks (CNNs). The model has been trained on a fictitious dataset that elaborates a model of multiple fraud scenarios and is also adaptable to modifications in the fraud scheme. Experimental results confirm the competitive advantage of the CNN model developed in terms of predictive power over traditional methods with a minimum number of False Positives.

Rupa Rani et al. [9] applied ML to develop a UPI fraud detection mechanism with XGBoost for high accuracy. SMOTE was used to solve the problem of class imbalance and to ensure proper representation of fraud cases. To enhance model performance, Principal Component Analysis (PCA) was used for feature selection and dimensionality reduction. The system monitors UPI transactions in real time using the predictive power of XGBoost and achieves 98.2% accuracy.

J Kavitha et al. [10] invoke a UPI fraud detection system that helps to detect and verify the normal behavior of transactions and anomalies by applying the Hidden Markov Model (HMM). The system incorporates K-means clustering, Auto Encoders, Local Outlier Factors (LOF), and Artificial Neural Networks (ANNs) to boost fraud detection. CNN is introduced for real-time detection of fraud addressing computational inefficiency and scalability.

Yash Patil et al. [11] introduced a UPI fraud detection system using ML techniques, particularly the Support Vector Machine (SVM) for analyzing user transactions, behaviors, and history of fraud cases. This system is ordinarily better than rule-based systems in anomaly detection and pattern recognition. However, such an SVM approach would raise challenges such as data imbalance, scalability, and interpretability of SVM models. The recommendation goes further and indicates that a certain case may be better by Random Forest (RF). Thus, it emphasizes feature selection, hybrid modeling, and continuous updates for effective performance.

Selvi P et al. [12] present a UPI fraud detection model with RF and AdaBoost algorithms. The study underlines the problematic imbalanced data, stressing different optimization techniques to improve their performance. The evaluation of models is done through various metrics to find the best way for fraud detection.

Vaishali Gupta et al. [13] proposed a DL-based fraud detection system for UPI transactions using Recurrent Neural Networks (RNNs), in particular, Long Short-Term Memory (LSTM) networks. The system aimed to identify complex transactional patterns over time that would improve fraud detection accuracy.

Jagadeesan et al. [14] propose the use of an RF algorithm to ensure better accuracy and minimal false positives for a UPI fraud detection system. The system uses supervised, unsupervised, and semi-supervised learning methodologies to detect anomalies in the transaction data. In addition, real-time monitoring and adaptive learning construct the system so that fraud detection becomes more adaptive to the modulating fraud patterns.

Sayalee S. Bodade et al. [15] have proposed a UPI fraud detection system using Decision Trees (DT), RF, Linear Regression (LR), and Gradient Boosting algorithms. The system performs real-time monitoring of transaction patterns, detects anomalies, and minimizes false positives. It is trained on the PaySim dataset using TRANSFER transactions, evaluating through multiple performance metrics to ensure accuracy.

Sumit M. Kulkarni et al. [16] propose a UPI fraud detection system that utilizes ML techniques such as RF, LR, DT, and SVM. SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Model-agnostic Explanations) make the model interpretable and transparent. Performance evaluations on several metrics verify that the algorithm RF demonstrated maximum accuracy with precision in detecting fraud and interpreting the model.

Shabreshwari et al. [17] presented an ML-based UPI fraud detection in which RF, LightGBM, XGBoost, DT, and LR are used. The system scans transaction data like amount, timestamp, payer-payee details, and device and location details for real-time fraud detection. The setup, integrated with alert mechanisms, adjudges performance using multiple metrics. The system has the intention to enhance UPI transaction security to recognize fraudulent patterns and reduce the risks related to financial fraud.

III. Methodology

A detailed methodology for detecting UPI transaction fraud is provided in this section. All the steps for data preprocessing, including data cleaning, encoding, and scaling, are described first, and then feature selection and imbalance handling are considered to refine the dataset. Finally, a robust ML model based on the XGBoost algorithm is trained, evaluated, and integrated into a user-friendly interface for real-time fraud detection.

Fig.1 shows the workflow of the proposed method presenting sequential steps from data collection and preprocessing, feature selection, data splitting, and addressing the class imbalance using SMOTE. Afterward, training is performed for model building with multiple algorithms evaluated based on key performance metrics. The last stage determines the best-performing model for maximum results.

A. Dataset Description

A full-fledged fraud detection system in UPI transactions is based on two distinct datasets. Dataset-1, with a size of 50,000 transactions, captures features such as transaction amount, frequency, payment gateway, Merchant Category, and time-based patterns, which are required for anomaly detection. Dataset-2, with 2,666 entries, is focused mainly on user-centric features such as transaction time, age, category, and fraud risk assessment variables for capturing behavioral insight into the activities of users. With the help of these integrated datasets, the model is equipped to study and identify fraudulent patterns originating from transaction data and user behavior, thus improving accuracy, as well as adaptability to emerging fraud methodologies.

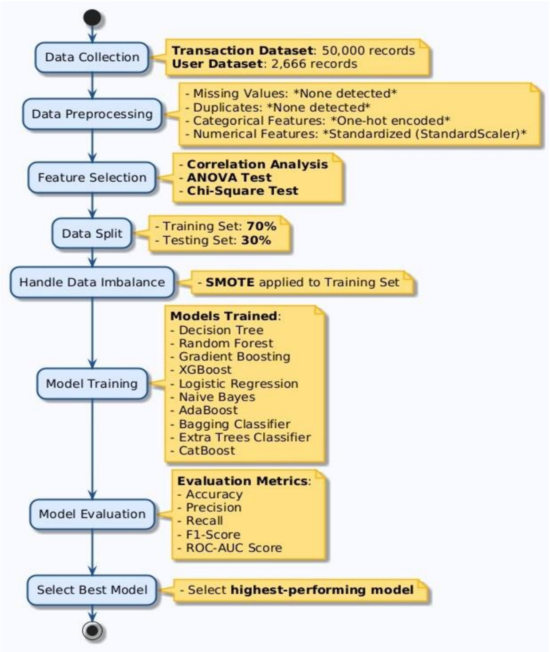


Fig. 1. Architectural Flow Diagram

B. Data Preprocessing

1) *Data Visualization:* The data visualization process focuses on understanding the distribution of fraudulent transactions through count plots. Correlation heatmaps help identify relationships between features. Visual representations like bar graphs and scatter plots assist in detecting patterns and anomalies in transaction data, enhancing fraud detection.

2) *Feature Selection:*

a) Correlation Analysis: It helps in identifying and removing highly correlated features that provide redundant information. By eliminating such features, the model avoids multicollinearity and improves performance.

$$r = \frac{\sum(p_i - \bar{p})(q_i - \bar{q})}{\sqrt{\sum(a_i - \bar{a})^2}\sqrt{\sum\left(b_i - \bar{b}\right)^2}} \tag{1}$$

b) ANOVA (Analysis of Variance) Test: It is used to assess whether there is a statistically significant difference between the means of numerical features across fraud and non-fraud classes. Features with high variance are retained for better fraud detection.

$$F = \frac{MS_B}{MS_W} = \frac{\frac{SS_B}{df_B}}{\frac{SS_W}{df_W}} \tag{2}$$

c) Chi-Square Test: This test evaluates the relationship between categorical features and the target variable. It helps in selecting features that have a strong association with fraudulent transactions.

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i} \tag{3}$$

3) *Encoding:* Encoding is crucial because ML models cannot directly process categorical data. One-hot encoding is applied to transform categorical features into numerical format. This eliminates the issue of the model interpreting categorical data as ordinal, thereby improving its performance in detecting fraudulent transactions.

4) *Scaling:* To bring numerical features to a common scale, StandardScaler is used to standardize the data by converting it to have a mean of 0 and a standard deviation of 1. This ensures that some powerful features do not dominate models, increasing the efficiency of LR and SVM.

5) *Data Imbalance Handling:* The data imbalance causes the model bias, and the effectiveness of fraud detection is decreased. SMOTE counteracts this through the generation of synthetic samples for the minority class so that overfitting could be avoided which will otherwise prevent good learning of the patterns. Consequently, the model becomes better capable of distinguishing between fraudulent and genuine transactions.

$$X_{new} = X_i + \lambda \times (X_j - X_i) \tag{4}$$

C. Model Selection

1) *Decision Tree:* It is an algorithm that splits data into decision nodes based on feature conditions; it is interpretable and handles both numerical and categorical data, important for identifying fraud patterns.

$$H(X) = -\sum_{j=1}^{n} P\left(X_j\right) \log_2 P\left(X_j\right) \tag{5}$$

2) *Logistic Regression:* This model is statistical and essentially uses a linear relationship among the attributes to create a probability estimator for fraud occurrence. It finds extensive applicability due to its simplicity and satisfactory efficiency concerning binary classification.

$$P(Z = 1|W) = \frac{1}{1 + e^{-(\alpha_0 + \alpha_1 W_1 + \cdots + \alpha_n W_n)}} \tag{6}$$

3) *Gaussian Naive Bayes (Gaussian NB):* Gaussian NB is a probabilistic classifier grounded upon Bayes' theorem, with an independence assumption on the set of features and normal (Gaussian) distribution for these features. It works fast and usually well on small datasets with continuous data.

$$P(V|U) = \frac{P(U|V)P(V)}{P(U)} \tag{7}$$

4) Random Forest: It is a type of EL technique that creates several branching trees and combines all of their predictions to make a final prediction with the aim of greater accuracy and less overfitting. Selected due to its robust nature and problem-solving techniques for imbalanced data.

$$\hat{g}(y) = \frac{1}{C}\sum_{c=1}^{C} g_c(y) \tag{8}$$

5) Gradient Boosting: Trees are boosted sequentially in Gradient Boosting according to the refinement of predictions by correcting errors. It's chosen to uncover complex fraud signals in UPI transactions. It performs well with imbalanced fraud data.

$$F_m(x) = F_{m-1}(x) + \gamma_m \sum_{i=1}^{N} g_i(x) \tag{9}$$

6) AdaBoost (Adaptive Boosting): AdaBoost combines several weak classifiers to produce a strong model whereby misclassified samples receive more focus. It has been used to amplify fraud detection accuracy on tricky UPI fraud cases and minimize false negatives.

$$w_i^{(t+1)} = w_i^{(t)} e^{-\alpha_t y_i h_t(x_i)} \tag{10}$$

7) Bagging Classifier: An EL method wherein several models are trained on different subsets of the data and average their predictions. It is selected to help reduce variance and improve stability.

$$\hat{g}(y) = \frac{1}{C}\sum_{c=1}^{C} g_c(y) \tag{11}$$

8) Extra Trees Classifier: One of the variations of the RF model is one that randomly selects points of split for more divergent trees, efficient in the complex feature set of UPI. Thus, it gives very fast and robust predictions for fraud.

$$H(Y) = -\sum_{i=1}^{n} P(Y_i)\, log_2\, P\,(Y_i) \tag{12}$$

9) CatBoost: This is a gradient-boosting method that evolves most intelligently in handling categorical features. It has been selected since UPI's categorical data like merchant type gives heavy weight to it. It is designed to get high accuracy with little tuning.

$$F_m(x) = F_{m-1}(x) + \gamma_m \sum_{i=1}^{N} g_i(x) \tag{13}$$

10) XGBoost: An advanced gradient boosting algorithm optimizing speed and performance is chosen for its efficiency with large-scale datasets, superiority in real-time fraud detection in UPI payments, and ability to effectively handle class imbalance.

$$L(\alpha) = \sum_{i=1}^{N} g(z_i, \hat{z_i}) + \sum_{k=1}^{K} \Omega(h_k) \tag{14}$$

D. Model Training

1) Dataset-1 (Hold-Out method): Considering the size and diversity of the dataset, the hold-out method is employed, dividing the data into 70% for training and 30% for testing. This allows efficient model evaluation on unseen data while preventing overfitting and reducing computational complexities.

$$D_{\text{train}} = \alpha D \tag{15}$$

$$D_{\text{test}} = (1 - \alpha)D \tag{16}$$

2) Dataset-2 (K-fold Cross-Validation): K-fold cross-validation is practiced for the lesser data set to maximize yield from scanty real data. K-subset the dataset, training on different folds. It reduces the effect of overfitting and gives a performance measure that is much more stable and unbiased. It is beneficial, especially for small datasets where there are few data points and each piece of information matters.

$$D_{\text{train}}^{(k)} = D \setminus D_{\text{test}}^{(k)} \tag{17}$$

$$D_{\text{test}}^{(k)} = D_k \tag{18}$$

The final evaluation score is computed as:

$$\text{Final score} = \frac{1}{K}\sum_{k=1}^{K} \text{Score}_\text{k} \tag{19}$$

IV. Results & Discussion

This section consists of a detailed evaluation of the fraud detection models using multiple performance metrics. Accuracy, Precision, Recall, F1-Score, ROC-AUC scores, and specificity were computed to ensure effective detection, while the confusion matrix and classification reports were employed to address the class imbalance and minimize errors.

TABLE I
PERFORMANCE SUMMARY OF MODELS

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	ROC AUC (%)	Specificity (%)
XGBClassifier	99.53	99.57	99.50	99.53	99.94	99.57
CatBoostClassifier	99.48	99.46	99.50	99.48	99.92	99.46
LogisticRegression	99.23	99.46	98.99	99.23	99.82	99.46
BaggingClassifier	99.07	98.83	99.32	99.07	99.91	98.82
DecisionTreeClassifier	98.59	98.45	98.74	98.59	99.16	98.44
RandomForestClassifier	96.55	98.34	94.70	96.49	99.56	98.40
GradientBoostingClassifier	93.49	96.58	90.18	93.27	98.45	96.80
AdaBoostClassifier	92.34	95.49	88.88	92.07	92.48	95.80
ExtraTreesClassifier	88.77	87.24	90.83	88.99	95.50	86.71
GaussianNB	84.99	99.39	70.41	82.43	94.53	99.57

Table I shows XGBoost with an accuracy of 99.53% stands out among all the models as having the best performance, while Gaussian NB, with an accuracy of 84.99%, is the least-performing model for Dataset-1.

For Dataset-2, similar behavior can be seen in that XGBoost has the highest accuracy of 96%, while the lowest-performing model is Logistic Regression with an accuracy of 82%.

Table II shows that the proposed model obtains an accuracy of 99.53%, which is higher than that of previous studies that provided different accuracy scores, with a 94% to 98.2% range. The true positive rate in one study was 87.5%; however, the false positive rate in that study was 13.4%. Thus, it indicates a very good tradeoff in detection performance levels. In sum, the proposed model shows a great improvement over existing systems.

TABLE II
COMPARISON OF MODEL PERFORMANCE WITH EXISTING STUDIES

Author	Performance
S. K. Lokesh Naik [7]	Accuracy: 97.68%
Rupa Rani [9]	Accuracy: 98.2%
Vaishali Gupta [13]	True Positive: 87.5%; False Positive: 13.4%
S. Jagadeesan [14]	Accuracy: 94%
Proposed Model	**Accuracy: 99.53%**

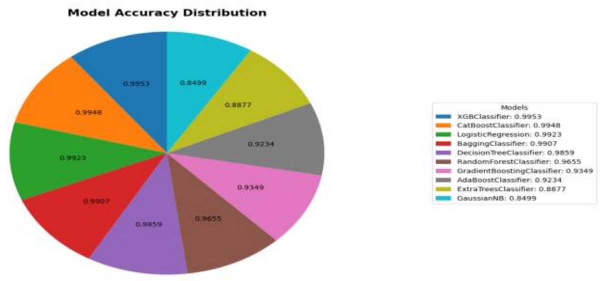


Fig. 2. Model Comparison based on Accuracy

The accuracy distributions for the different ML models in Fig. 2 give an insight into their performance variation. From the plot, it can be observed that some models yield higher accuracy, while others are relatively low in performance. This visualization helps in determining the best models for any given task based on their accuracy.

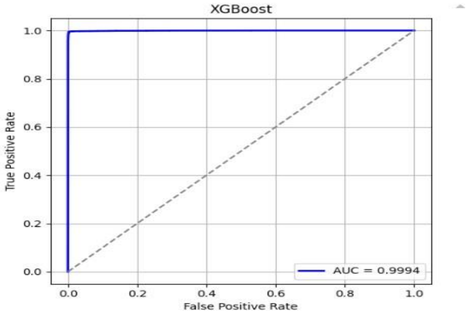


Fig. 3. AUC Curve of Highly Performed Model (XGBoost)

The curve in Fig. 3 represents that the True Positive Rate (TPR) is nearing 1 at different False Positive Rates (FPR), with an AUC value of 0.9994, which reflects extremely good model performance. Such a high AUC indicates the quality of the model XGBoost in separating both positive and negative classes with minimum errors.

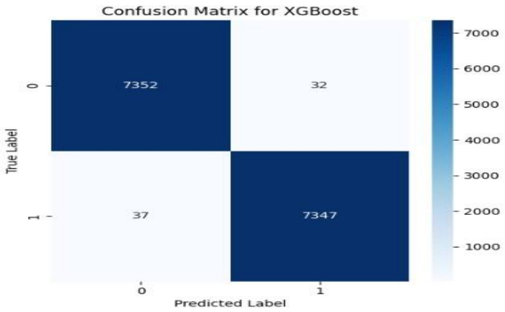
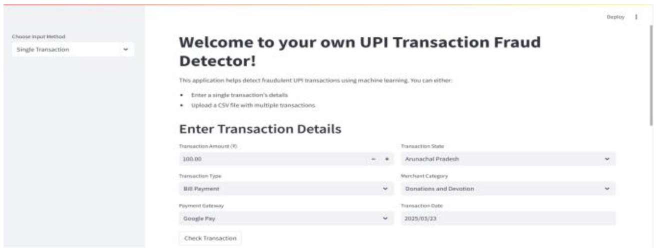


Fig. 4. Confusion Matrix of Highly Performed model (XGBoost)

Fig. 4 shows the confusion matrix for the XGBoost model, indicating that it has 7352 true negatives and 7347 true positives, which is very accurate. There are only 32 false positives and 37 false negatives; therefore, misclassifications are minimal. This confirms the excellent performance of XGBoost in distinguishing fraudulent from non-fraudulent transactions.

XGBoost excels in fraud detection by effectively handling imbalanced datasets and preventing overfitting through regularization techniques. Its sequential tree-building process, where each tree corrects the errors of its predecessors using gradient descent to optimize a loss function, enhances precision and recall—key metrics for identifying fraud. Hyperparameter tuning further refines its performance. Built-in cross-validation and feature importance analysis provide valuable insights for model validation. With computational efficiency and scalability, XGBoost suits large datasets, enabling rapid processing. These strengths make it a robust tool for real-time fraud prevention.



Choose Input Method

Welcome to your own UPI Transaction Fraud Detector!

This application helps detect fraudulent UPI transactions using machine learning. You can either:

- Enter a single transaction's details
- Upload a CSV file with multiple transactions

Enter Transaction Details

Transaction Amount (₹): 100.00

Transaction State: Arunachal Pradesh

Transaction Type: Bill Payment

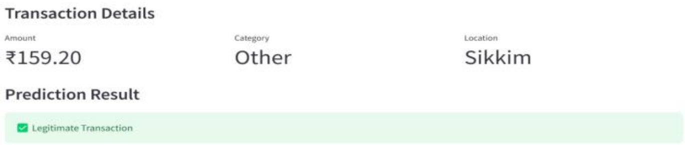
Merchant Category: Donations and Devotion

Payment Gateway: Google Pay

Transaction Date: 2025/03/23

Fig. 5. Streamlit-based Interface for UPI Fraud Detection System

Fig. 5 shows an interface of the web application which is designed using Streamlit to detect fraudulent UPI transactions wherein users can input one-time or bulk transactions for checking possible fraudulent activity through ML methods. It has provisions for entering the transaction amount, state, type, merchant category, payment gateway, and date. It has a button "Check Transaction" for running fraud detection analysis.



Transaction Details

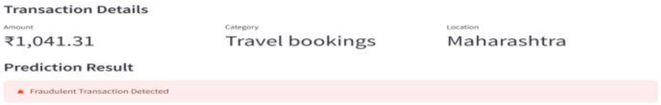
Amount	Category	Location
₹159.20	Other	Sikkim

Prediction Result

✓ Legitimate Transaction

Fig. 6. Legitimate Transaction

Fig. 6 shows a UPI fraud detection result indicating transaction details that include amount, category, and location. The system predicts the transaction to be genuine as indicated by a green checkmark.



Transaction Details

Amount
₹1,041.31

Category
Travel bookings

Location
Maharashtra

Prediction Result

▲ Fraudulent Transaction Detected

Fig. 7. Fraud Transaction

Fig. 7 displays a UPI fraud detection result for a transaction related to travel bookings in Maharashtra. The system flags it as fraudulent, highlighted with a red warning message.

V. Conclusion

This research highlights the significance of using ML tools especially EL techniques like XGBoost and CatBoost in identifying fraudulent UPI transactions. The research employed thorough data preprocessing, feature engineering, and strict model evaluation to achieve an outstanding accuracy rate of more than 99% with minimal misclassification errors. The optimized XGBoost model could be considered a scalable and real-time solution ready for implementation by financial institutions to enhance fraud prevention mechanisms. Moreover, the model comes with a user-friendly interface that increases the practical implementation of the system by bridging the gap between an advanced technical model and the end users such as fraud analysts, and financial professionals. This interface ensures the integration of the system into the existing workflow and enables real-time fraud detection and improved operational efficiency. On the other side, the results make digital payment platforms more trustworthy and have proved that there is a synergy between the credibility of advanced analytics and user-centric designs for safeguarding digital transactions. Future work could extend to using Blockchain in UPI fraud detection for a tamper-proof ledger and incorporate a smart contract system, possibly even DL and behavioral biometrics, to improve accuracy and fight even highly sophisticated fraud tactics.

References

[1] EBANX. (2025, March 15). *Unified Payments Interface (UPI) explained*. https://insights.ebanx.com/en/resources/payments-explained/ unified-payments-interface-upi/

[2] National Payments Corporation of India. (2025, March 17). *UPI product overview.* https://www.npci.org.in/what-we-do/upi/product-overview

[3] RazorPay. (2025, March 24). *What is UPI and how it works?* https://razorpay.com/blog/what-is-upi-and-how-it-works/

[4] CNBC TV18. (2024, November 25). *UPI fraud cases surge by 85% in FY24: Key insights and data.* https://www.cnbctv18.com/business/finance/upi-fraud-cases-rise-85-pcin-fy24-increase-parliament-reply-data-19514295.htm

[5] Pragmatic Coders. (2023, November 3). *Machine learning for fraud detection in FinTech.* https://www.pragmaticcoders.com/blog/machinelearning-for-fraud-detection-in-fintech

[6] Pan, E. (2024). Machine Learning in Financial Transaction Fraud Detection and Prevention. *Transactions on Economics, Business and Management Research, 5*, 243-249. https://doi.org/10.62051/16r3aa10

[7] Lokesh Naikl, S. K., Kiran, A., Kumar, V. P., Mannam, S., Kalyani, Y., & Silparaj, M. (2024, March). Fraud fighters—How AI and ML are revolutionizing UPI security. *Proceedings of the IEEE International Conference on Science, Technology, Engineering, and Management (ICSTEM)*. https://doi.org/10.1109/ICSTEM61137.2024.10560740

[8] Bello, O. A., & Olufemi, K. (2024). Artificial intelligence in fraud prevention: Exploring techniques, applications, challenges, and opportunities. *Computer Science & IT Research Journal, 5*(6), 1505–1520. https://doi.org/10.51594/csitrj.v5i6.1252

[9] Rani, R., Alam, A., & Javed, A. (2024). Secure UPI: Machine learning-driven fraud detection system for UPI transactions. In *Proceedings of the 2nd International Conference on Disruptive Technologies (ICDT)* (pp. 924–928). IEEE. https://doi.org/10.1109/ICDT61202.2024.10489682

[10] Kavitha, J., Indira, G., Kumar, A., Shrinita, A., & Bappan, D. (2024). Fraud detection in UPI transactions using ML. *EPRA International Journal of Research & Development (IJRD)*, 9(4), 142-146. https://doi.org/10.36713/epra16459

[11] Patil, Y., Shinde, A., Parthe, Y., & Sayyad, S. (2024, September). UPI fraud detection using machine learning. *International Research Journal of Modernization in Engineering Technology and Science*, 6(9), 142–146.

[12] Selvi, P., & Suryadharshan, S. (2024). UPI fraud detection using machine learning. *International Journal of Innovative Research in Computer and Communication Engineering, 12*(6), 8764–8768.

[13] Gupta, V., Sharma, S., Nimkar, S., & Pathak, S. (2024, March). UPI-based financial fraud detection using deep learning approach. In *2024 International Conference on Advances in Computing Research on Science Engineering and Technology (ACROSET)*. IEEE. https://doi.org/10.1109/ACROSET62108.2024.10743663

[14] Jagadeesan, S., Arjun, K. S., Dhanika, G., Karthikeyan, G., and Deepika, K. (2024, March). UPI fraud detection using machine learning. In V. Sharmila et al. (Eds.), *Challenges in Information, Communication and Computing Technology* (pp. 755–760). Taylor & Francis. https://doi.org/10.1201/9781003559085-130

[15] Bodade, S. S., & Pawade, P. P. (2024). Implementation paper on UPI fraud detection using machine learning. *Journal of Emerging Technologies and Innovative Research*, *11*(4), c947–c954. http://www.jetir.org/papers/JETIR2404299.pdf

[16] Kulkarni, S. M., & Gavhane, J. (2024). UPI fraud detection using machine learning. *International Journal of Creative Research Thoughts (IJCRT)*, *12*(12), i169–i174. http://www.ijcrt.org/papers/IJCRT2412897.pdf

[17] Shabreshwari, R. M., Mehrooz, S., Fatima, S., Tanmai, R. B., & Manasali, G. (2024). UPI fraud detection using machine learning. *International Journal of Advances in Engineering and Management (IJAEM)*, *6*(6), 98–100.