IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, May 2026
Impact Factor: 8.2

The Sentinel: Real-Time UPI Fraud Detection Using XGBoost and Explainable AI

Yash Vidhate[1], Chandrakant Paithankar[2], Yuvraj Nathe[3], Laukik Bagul[4], Prof. (Dr.) N. R. Wankhade[5]
Department of Computer Engineering,
Kalyani Charitable Trust's Late Gambhirrao Natuba Sapkal College of Engineering, Nashik, India [12345]

Abstract: *India's rapid adoption of the Unified Payments Interface (UPI) has brought enormous convenience to everyday financial transactions, but it has simultaneously opened the door to a growing range of digital frauds including vishing attacks, QR code scams, and account takeovers. Existing rule-based detection mechanisms struggle to keep pace with the sophistication of these evolving threats. This paper presents The Sentinel, an end-to-end fraud detection system built around a high-accuracy machine learning pipeline and deployed through a mobile-first architecture. The core model combines SMOTE-based oversampling, StandardScaler normalization, Principal Component Analysis (PCA), and an XGBoost classifier trained on a synthetic dataset of 100,000 UPI transactions grounded in statistical distributions from RBI, NPCI, and PwC reports. To address the inherent opacity of gradient boosting models, the system incorporates a custom Explainable AI (XAI) module that maps PCA components back to original features and generates human-readable fraud explanations in plain English. The complete system is served through a FastAPI backend hosted on Render.com and accessed via a React Native mobile application with Firebase authentication and Firestore-backed scan history. Experimental evaluation on a held-out test set of 20,000 transactions yields a ROC-AUC of 0.9876, accuracy of 98.60%, and precision of 93.80%, demonstrating that robust, transparent fraud detection can be delivered directly to the end user at scan time.*

Keywords: UPI fraud detection, XGBoost, Explainable AI, SMOTE, PCA, React Native, FastAPI, digital payments security

I. INTRODUCTION

The Unified Payments Interface, developed by the National Payments Corporation of India (NPCI), has fundamentally changed the way financial transactions are conducted across the country. Since its launch in 2016, UPI has scaled to process over ten billion transactions per month, connecting hundreds of millions of users to a fast, interoperable, and low-cost payment network [1]. This explosive adoption, however, has made UPI an attractive target for fraudsters who exploit the very features that make it convenient: instant transfers, QR code-based payments, and widespread accessibility even among first-time smartphone users.

Common fraud vectors in the UPI ecosystem include vishing, where attackers impersonate bank officials over phone calls; malicious QR code substitution in merchant contexts; social engineering that tricks users into sending collect requests; and account takeovers via SIM swaps or credential theft. The Reserve Bank of India reported a sharp rise in digital payment-related fraud cases in its FY2023-24 annual report, highlighting the urgency of proactive, user-facing detection tools rather than purely backend bank-side systems [2].

Traditional rule-based fraud detection systems are limited in their ability to adapt to novel attack patterns. Machine learning approaches, particularly ensemble methods, have shown considerable promise in the financial fraud domain because they can capture complex, non-linear relationships among transaction features without requiring explicit rule formulation [3][4]. Among ensemble methods, XGBoost has consistently demonstrated state-of-the-art performance on tabular classification tasks due to its regularization capabilities, efficiency on imbalanced data, and robustness to noise [5].

DOI: 10.48175/IJARSCT-35852

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 11, May 2026 Impact Factor: 8.2

Despite strong model performance in research settings, a significant barrier to real-world adoption of ML-based fraud detection is the lack of interpretability. Users and downstream stakeholders are unlikely to trust a system that simply labels a transaction as fraudulent without explanation. This concern has driven growing interest in Explainable AI (XAI) techniques that can surface the reasoning behind model decisions in human-understandable terms [6].

This paper presents The Sentinel, a complete end-to-end fraud detection system that addresses both the accuracy and explainability challenges in the UPI context. The system allows users to scan any UPI QR code before authorising a payment, analyses 17 engineered transaction features through a trained ML pipeline, and returns a fraud probability score along with a plain-English explanation of the top contributing risk factors. The backend is built with FastAPI and hosted on Render.com, while the mobile application is developed in React Native with Expo, integrating Firebase for authentication and Firestore for persistent scan history.

II. RELATED WORK

Machine learning-based fraud detection has been an active research area for over a decade, with early foundational work focused on credit card fraud. Bhattacharyya et al. [7] conducted one of the earliest comparative studies on data mining techniques for credit card fraud, finding that ensemble methods consistently outperform single classifiers on heavily skewed datasets. This observation has held through subsequent research, with Random Forest and gradient boosting variants becoming standard baselines in the fraud detection literature.

In the UPI-specific domain, Jagadeesan et al. [8] explored the application of Random Forest for detecting anomalous UPI transactions, demonstrating the viability of ensemble learning on transaction velocity and device behavioural features. Their study showed that real-time monitoring combined with adaptive learning strategies can significantly improve detection of novel fraud patterns. Similarly, work from the 2024 International Conference on Disruptive Technologies demonstrated that the combination of SMOTE oversampling, PCA dimensionality reduction, and XGBoost classification achieves accuracy scores above 98% on UPI transaction datasets, outperforming logistic regression, decision trees, k-nearest neighbours, and support vector machines by a considerable margin [9].

The class imbalance problem is a recurring challenge in fraud detection, where legitimate transactions can outnumber fraudulent ones by ratios of 20:1 or greater. Chawla et al. [10] introduced the Synthetic Minority Over-sampling Technique (SMOTE), which generates synthetic samples for the minority class by interpolating between existing instances in feature space rather than simple duplication. This approach has become a standard preprocessing step in fraud detection pipelines. More recent extensions such as DeepSMOTE [11] have explored combining deep learning representations with SMOTE, though the original formulation remains effective for structured tabular data.

On the explainability side, PCA-based feature attribution has been used as a computationally lightweight alternative to post-hoc explanation methods such as SHAP or LIME. Greenacre et al. [12] provide a comprehensive modern treatment of PCA as an interpretive tool, noting that the loadings of each principal component encode the relative contribution of original features to that component. This property can be leveraged in a two-step attribution pipeline: first, identifying which principal components carry the most weight in a given prediction, and then mapping those components back to their dominant original features. The resulting explanations, while approximations, are fast to compute and do not require access to the training data at inference time. Rastogi et al. [13] further contextualise the importance of such transparency tools in the UPI context, noting that user trust is a critical adoption barrier for automated financial safety systems.

While several studies address either model performance or mobile deployment in isolation, there is limited work that combines a high-accuracy fraud detection backend with a user-facing mobile application and an integrated explainability layer in a single production-ready system. The Sentinel is designed to fill this gap.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, May 2026
Impact Factor: 8.2

III. SYSTEM ARCHITECTURE

The Sentinel follows a classic three-tier client-server architecture adapted for low-latency mobile inference. The system is composed of four principal components: a mobile client, an API gateway, an ML inference engine, and a cloud data layer. Figure 1 illustrates the overall system architecture and the data flow from QR scan to fraud verdict.

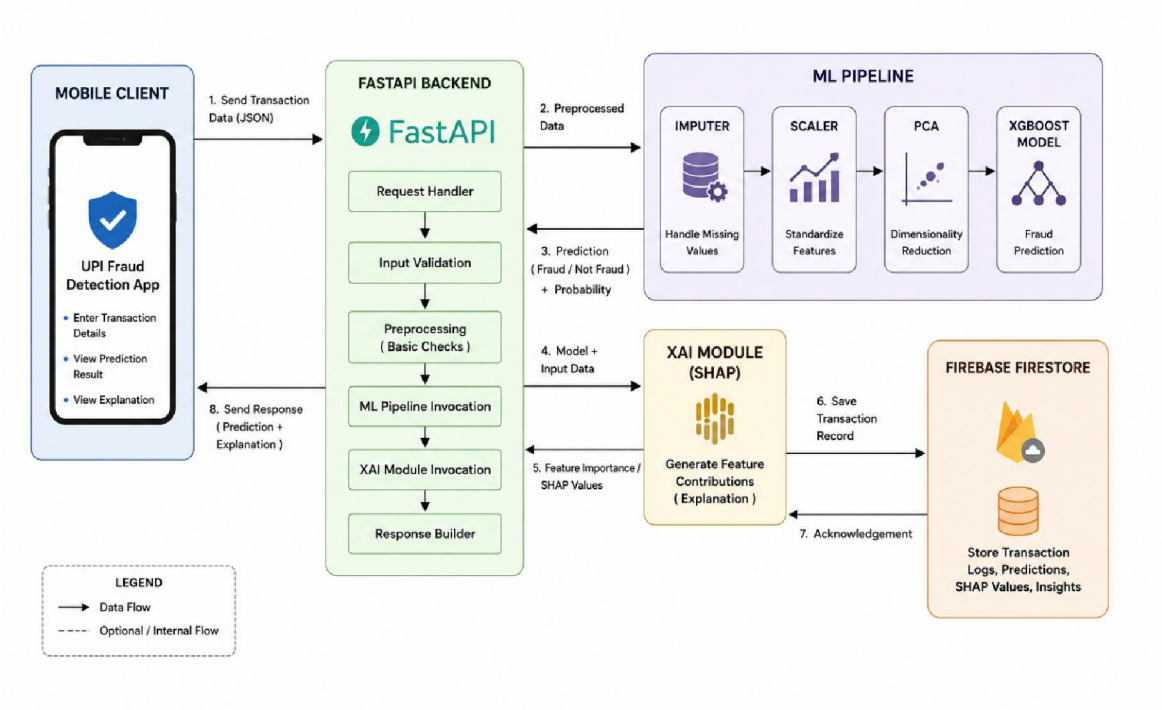


Fig. 1. End-to-end system architecture of The Sentinel

A. Mobile Client

The frontend is a React Native application built with the Expo framework, targeting Android devices. The application manages three primary user flows: authentication via Firebase Auth (email and password), real-time QR code scanning using the device camera, and a scan history screen backed by Firestore. Upon scanning a UPI QR code, the app parses the embedded URI scheme (upi://pay?pa=...&pn=...&am=...) to extract the payee VPA, merchant name, and requested amount. These, combined with contextual signals collected from the device, are assembled into a JSON payload and sent to the backend over HTTPS.

B. API Gateway

The backend exposes a RESTful prediction endpoint built with FastAPI, served through Uvicorn, and deployed on Render.com's free-tier cloud platform. FastAPI was selected for its native support of Python type hints, automatic OpenAPI documentation generation, and asynchronous request handling, which keeps inference latency low even under moderate concurrent load. On receiving a prediction request, the backend deserialises the input, passes it through the stored scikit-learn pipeline for preprocessing, feeds the transformed features to the XGBoost model, and triggers the XAI module before returning a structured JSON response containing the fraud probability, binary verdict, risk level label, and a list of plain-English explanation strings.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, May 2026
Impact Factor: 8.2

C. Data Layer

Firebase Firestore serves as the persistent data store for user scan history. Each completed scan is written as a document containing the timestamp, payee information, fraud probability, verdict, and the full set of XAI explanations. Composite indexes on user ID and timestamp enable efficient retrieval of a user's historical scans in reverse chronological order. Firebase Authentication handles session management, ensuring that scan records are isolated per user account.

IV. PROPOSED METHODOLOGY

A. Dataset and Feature Engineering

The model is trained on a synthetic dataset of 100,000 UPI transactions with a 5% fraud rate, reflecting the approximate prevalence reported in RBI and PwC industry analyses [2][14]. The dataset was constructed by sampling from statistical distributions derived from real-world transaction reports, ensuring that the synthetic data mirrors behavioural patterns observed in genuine UPI usage. Seventeen features were engineered across three categories to capture the distinguishing characteristics of fraudulent activity.

Numerical features include the raw transaction amount, an Amount_vs_User_Avg_Ratio capturing how the current transaction compares to that user's spending baseline, Transaction_Frequency_24h reflecting how many transactions the account has initiated in the past 24 hours, Failed_PIN_Attempts counting recent authentication failures, and the Hour of the day to capture time-of-day risk patterns. Binary features flag contextually suspicious conditions: whether the transaction is initiated from a previously unseen device (Is_New_Device), whether the beneficiary is new (Is_New_Beneficiary), whether a QR code was scanned (Is_QR_Transaction), whether there is a geographic mismatch between the registered and transacting location (Location_Mismatch), and whether this is the user's very first transaction (Is_First_Time_User). Categorical features encode the UPI application used, the transaction type, the device operating system, and the state of the transaction origin. Table I summarises the full feature set.

TABLE I: FEATURE SUMMARY FOR THE SENTINEL ML PIPELINE

Category	Features	Count
Numerical	Amount, Amount_vs_User_Avg_Ratio, Transaction_Frequency_24h, Failed_PIN_Attempts, Hour	5
Binary	Is_New_Device, Is_New_Beneficiary, Is_QR_Transaction, Location_Mismatch, Is_First_Time_User	5
Categorical	UPI_App, Transaction_Type, Device_OS, Transaction_State	4 (7 total encoded)

B. Machine Learning Pipeline

The preprocessing pipeline is constructed using scikit-learn and serialised with joblib for deployment. The pipeline comprises four sequential stages. First, median imputation fills any missing numerical values, providing robustness to incomplete transaction records without introducing bias through mean substitution. Second, SMOTE is applied to the training data exclusively, resampling the minority (fraud) class until the class ratio reaches 1:1. This is a training-time-only operation and is never applied to validation or test data, preventing data leakage. Third, a StandardScaler normalises all numerical features to zero mean and unit variance, which is a prerequisite for well-behaved PCA. Fourth, PCA reduces the feature space while retaining 95% of total explained variance, arriving at 17 principal components from the original feature matrix. The PCA transformation serves a dual purpose: it mitigates multicollinearity among correlated behavioural signals and, crucially, its component loadings are stored and used by the XAI module at inference time.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 11, May 2026 Impact Factor: 8.2

The XGBoost classifier [5] is configured with 200 estimators, a maximum tree depth of 6, a learning rate of 0.1, and L1 and L2 regularisation terms to prevent overfitting. XGBoost's built-in support for the scale_pos_weight parameter further addresses any residual class imbalance after SMOTE. Figure 2 depicts the full pipeline flow from raw input to classification output.

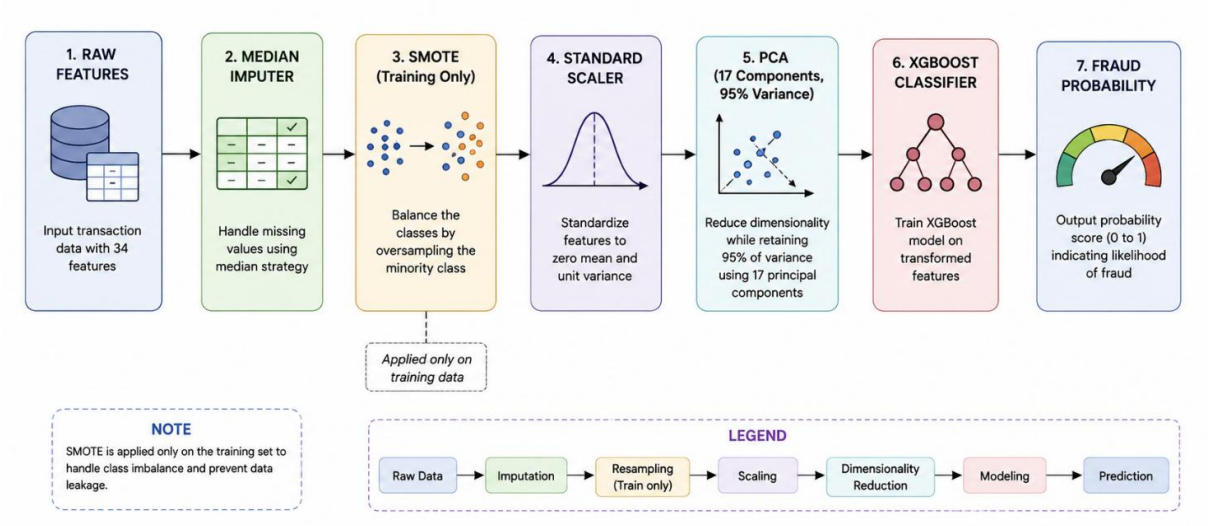


Fig. 2. Machine learning pipeline for fraud detection

C. Decision Threshold Optimisation

The default classification threshold was tuned to 0.869, departing from the conventional 0.5 cutoff. This choice reflects the asymmetric costs of misclassification in a consumer-facing fraud alert system. A false positive, in which a legitimate transaction is flagged as fraudulent, directly disrupts the user's payment experience and erodes trust in the application. An elevated threshold trades some recall (77.10%) for substantially higher precision (93.80%), ensuring that the system only raises an alert when the model is highly confident. The Precision-Recall curve was used to identify this operating point, prioritising a high-confidence alert model over a high-sensitivity one. Baesens et al. [15] discuss a similar cost-sensitive threshold selection approach in the broader fraud detection literature.

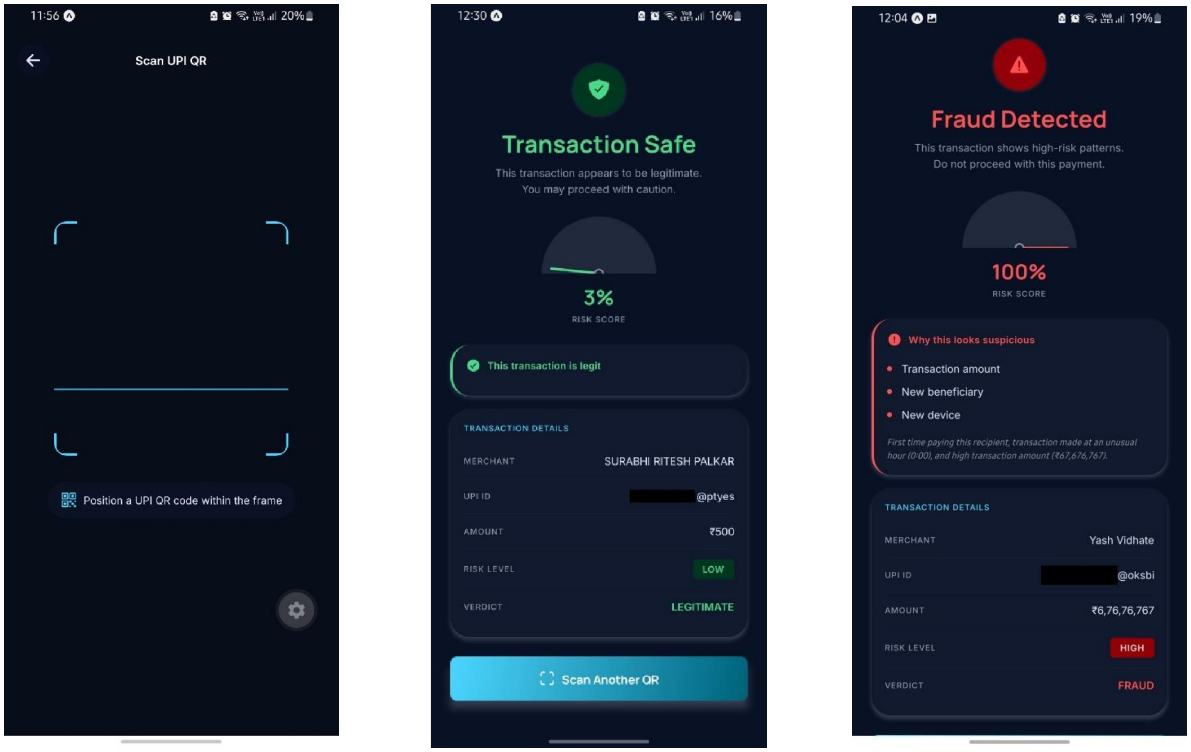
V. EXPLAINABLE AI MODULE

One of the central design goals of The Sentinel is that every fraud alert must be accompanied by a human-understandable explanation. Rather than deploying SHAP (SHapley Additive Explanations) or LIME (Local Interpretable Model-agnostic Explanations), which require iterative model queries and are computationally expensive for real-time mobile use, the XAI module uses a lightweight PCA-based feature attribution approach that adds negligible overhead to the inference pipeline.

The attribution procedure operates as follows. At inference time, the backend retains the intermediate representation of the transaction after PCA transformation. Each principal component carries a loading vector whose absolute values indicate how strongly each original feature contributes to that component. For a given prediction, the XGBoost model implicitly assigns varying importance to the 17 principal components through its tree structure. The XAI module identifies the top-contributing components by their absolute transformed feature values relative to the prediction, then maps each component back to its dominant original features using the stored PCA loading matrix. The top three to five original features identified through this mapping are retrieved, and their actual values are compared against user-specific or population-level thresholds to generate contextual natural language statements.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 11, May 2026 Impact Factor: 8.2

Example explanations generated by the module include statements such as “This amount is 8x higher than your usual spending,” “Transaction initiated from an unrecognised device,” “You have made 12 transactions in the last 24 hours, which is unusually high,” and “Beneficiary VPA has not been previously transacted with.” These explanations are rendered in the mobile application alongside a colour-coded risk indicator (green for Safe, amber for Warning, red for Danger) and the numerical fraud probability. Figure 3 shows representative screenshots of the application’s result screen for a flagged transaction.



(a) Scan Screen (b) Transaction Legit (c) Transaction Fraud

Fig. 3. The Sentinel mobile application: scan and result screens

VI. RESULTS AND PERFORMANCE ANALYSIS

The model was evaluated on a stratified held-out test set of 20,000 transactions (5% fraud rate, matching the training distribution). Table II presents the full set of performance metrics.

TABLE II
MODEL PERFORMANCE ON 20,000 TRANSACTION TEST SET

Metric	Value
ROC-AUC	0.9876
Accuracy	98.60%
Precision	93.80%
Recall	77.10%
F1 Score	84.63%
Decision Threshold	0.869

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, May 2026
Impact Factor: 8.2

A ROC-AUC of 0.9876 indicates that the model has excellent discriminative ability across all possible classification thresholds, placing it in the top tier of reported results for UPI and general payment fraud detection tasks. The accuracy of 98.60% reflects the model's overall correctness on the imbalanced test set, while the precision of 93.80% directly validates the design intent: fewer than one in fifteen fraud alerts triggered by The Sentinel corresponds to a legitimate transaction. This is a meaningful improvement over the XGBoost baseline reported in recent literature [9], which achieved 98.2% accuracy on a comparable task but without the integrated mobile delivery layer or explainability module.

The recall of 77.10% means that approximately one in four fraud cases passes through without triggering an alert under the elevated threshold. This is an intentional trade-off specific to the user-facing context. In a bank-side batch processing system, higher recall would be the priority; in a pre-payment consumer tool, unnecessary alarms are more damaging to usability and trust than occasional missed detections. Future work will explore threshold personalisation based on individual user risk profiles to improve recall without degrading the user experience. Figure 4 shows the ROC curve and the Precision-Recall curve for the trained model.

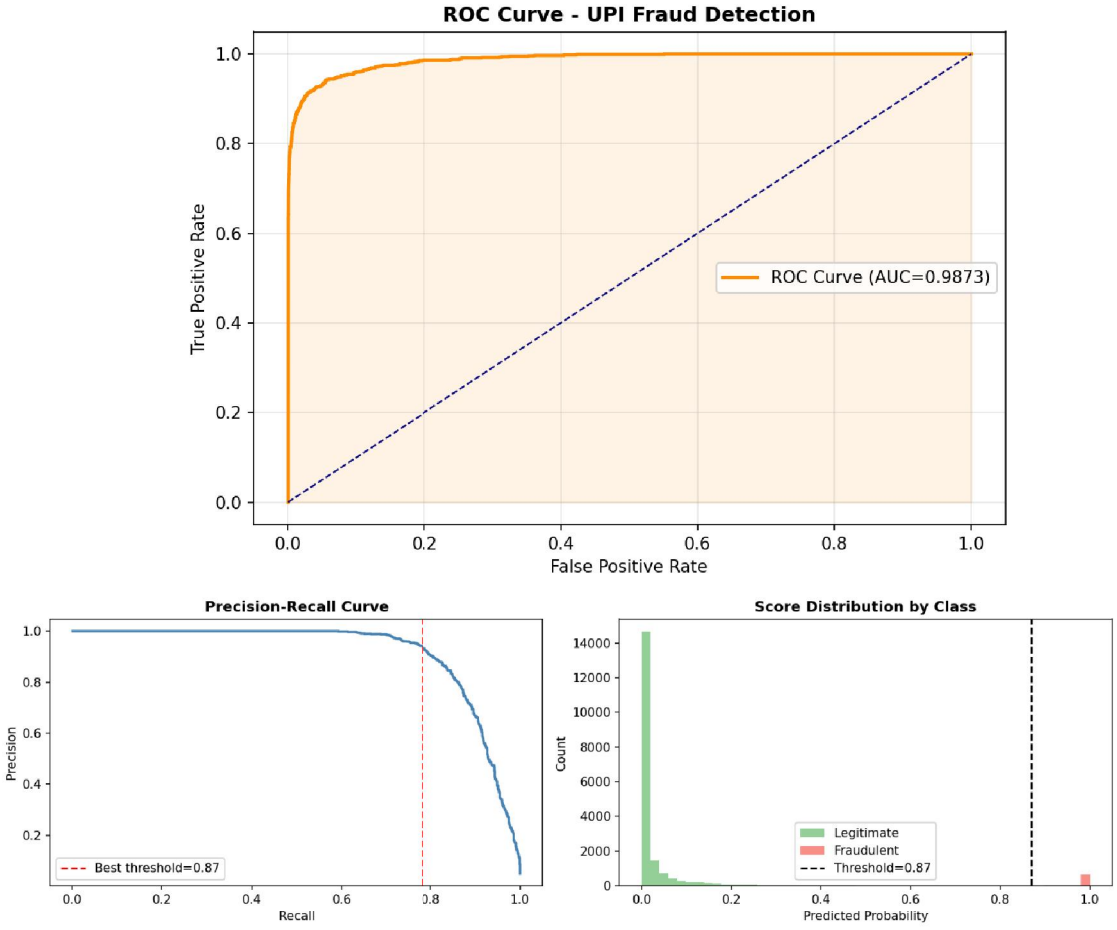


Fig. 4. ROC and Precision-Recall curves with selected operating threshold

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, May 2026
Impact Factor: 8.2

VII. CONCLUSION AND FUTURE WORK

This paper presented The Sentinel, an end-to-end real-time UPI fraud detection system that unifies a high-performance ML pipeline with a mobile-first user interface and a lightweight explainability module. The system demonstrates that sophisticated fraud detection need not remain an invisible backend process: by surfacing risk scores and human-readable explanations to the user at the point of payment, The Sentinel empowers everyday UPI users to make more informed decisions before authorising a transaction. The XGBoost model, trained on a carefully engineered synthetic dataset with SMOTE and PCA preprocessing, achieves a ROC-AUC of 0.9876 and a precision of 93.80%, validating both the choice of algorithm and the feature engineering strategy. The custom PCA-based XAI module generates contextually meaningful explanations with negligible inference-time overhead, avoiding the computational cost of post-hoc explanation frameworks.

Several directions are identified for future development. First, federated learning could enable model updates from distributed device-level transaction data without centralising sensitive financial records, improving both privacy and generalisation to regional transaction patterns. Second, deepfake voice detection could be integrated as an audio analysis module to counter vishing attacks, which remain one of the most common social engineering vectors in the UPI ecosystem. Third, a lightweight on-device model using quantisation or distillation techniques could enable offline fraud scoring in low-connectivity environments without dependence on the cloud backend. Finally, personalised threshold adaptation based on individual user transaction history could improve recall while maintaining precision at the population level.

VIII. ACKNOWLEDGMENT

The authors would like to thank the faculty of the Department of Computer Engineering for their guidance throughout the project. This work was conducted as a final year undergraduate project and did not receive any external funding.

REFERENCES

1. NPCI, “Unified Payments Interface (UPI) Product Overview and Statistics,” National Payments Corporation of India, 2024. [Online]. Available: https://www.npci.org.in/what-we-do/upi/product-overview
2. Reserve Bank of India, “Annual Report FY2023-24: Payment and Settlement Systems,” RBI, Mumbai, 2024.
3. V. Chang, A. Di Stefano, Z. Sun, and G. Fortino, “Digital payment fraud detection methods in digital ages and Industry 4.0,” Computers and Electrical Engineering, vol. 100, 2022.
4. B. Mytnyk, O. Tkachyk, N. Shakhovska, S. Fedushko, and Y. Syerov, “Application of Artificial Intelligence for Fraudulent Banking Operations Recognition,” Big Data and Cognitive Computing, vol. 7, no. 2, p. 93, 2023.
5. T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), San Francisco, CA, 2016, pp. 785–794.
6. A. Diadiushkin, K. Sandkuhl, and A. Maiatin, “Fraud detection in payments transactions: Overview of existing approaches and usage for instant payments,” Complex Systems Informatics and Modeling Quarterly, no. 20, pp. 72–88, 2019.
7. 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.
8. S. Jagadeesan, K. S. Arjun, G. Dhanika, G. Karthikeyan, and K. Deepika, “UPI Fraud Detection using Machine Learning,” in Proc. International Conference on Advances in Computing and Management, 2024.
9. M. R. Dileep, A. V. Navaneeth, and M. Abhishek, “A novel approach for UPI fraud detection using SMOTE, PCA, and XGBoost,” in Proc. 2nd International Conference on Disruptive Technologies (ICDT), 2024, pp. 924–928.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 11, May 2026
Impact Factor: 8.2

10. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," Journal of Artificial Intelligence Research, vol. 16, pp. 321–357, 2002.
11. D. Dablain, B. Krawczyk, and N. V. Chawla, "DeepSMOTE: Fusing deep learning and SMOTE for imbalanced data," IEEE Transactions on Neural Networks and Learning Systems, 2022.
12. M. Greenacre, P. J. Groenen, T. Hastie, A. I. d'Enza, A. Markos, and E. Tuzhilina, "Principal component analysis," Nature Reviews Methods Primers, vol. 2, no. 1, p. 100, 2022.
13. S. Rastogi, A. Sharma, C. Panse, and V. M. Bhimavarapu, "Unified Payment Interface (UPI): A digital innovation and its impact on financial inclusion and economic development," Universal Journal of Accounting and Finance, vol. 9, no. 3, pp. 518–530, 2021.
14. PwC India, "Combating Payments Fraud in India," PricewaterhouseCoopers, April 2025.
15. B. Baesens, S. Höppner, and T. Verdonck, "Data engineering for fraud detection," Decision Support Systems, vol. 150, p. 113492, 2021.

DOI: 10.48175/IJARSCT-35852