© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716166

Detection of Mule Accounts and Fraudsters in UPI Transactions Using AI and Machine Learning Techniques

SIBA SAHU[1], PRIYANKA CHAUDHURY[2], SIBA PRASAD SENAPATI[3], SUBHAKANTA PRADHAN[4], SUNIL KUMAR NAHAK[5]

[1,2,3,4] *B. Tech 4th Year Students, Department of Computer Science & Engineering*
NIST University, Berhampur, India

[5]*Assistant Professor, Department of Computer Science & Engineering*
NIST University, Berhampur, India

Abstract- ***The rapid expansion of the Unified Payments Interface (UPI) ecosystem in India has been accompanied by a significant rise in digital financial fraud, particularly through the use of mule accounts - bank accounts used by fraudsters to launder illicitly obtained funds. Conventional rule-based fraud detection systems are increasingly inadequate against the sophistication and volume of modern UPI fraud. This paper proposes a comprehensive multi-model machine learning framework that integrates Gradient Boosted Decision Trees (GBDT), Graph Neural Networks (GNN), and Long Short-Term Memory (LSTM) networks to detect mule accounts and fraudulent actors in UPI transaction data. The proposed system analyses transaction behaviour, network topology of fund flows, and temporal transaction patterns to accurately identify suspicious accounts. Experimental results demonstrate a detection accuracy of 94.3%, precision of 92.7%, recall of 91.8%, and an F1-score of 0.923, outperforming existing single-model approaches. The framework offers a scalable, interpretable, and real-time deployable solution for banks, payment service providers, and financial regulators***

Index Terms- ***Mule Account Detection, UPI Fraud, Unified Payments Interface, Graph Neural Networks, LSTM, Gradient Boosted Trees, Financial Fraud Detection, Digital Payments, Anomaly Detection, Anti-Money Laundering, Transaction Monitoring, Machine Learning***

I. INTRODUCTION

India's Unified Payments Interface (UPI), launched by the National Payments Corporation of India (NPCI) in 2016, has transformed the digital payments landscape. With over 13 billion monthly transactions recorded in 2024 and a total value exceeding ₹20 lakh crore annually, UPI has become the backbone of India's digital economy. However, this unprecedented scale of adoption has simultaneously created fertile ground for financial fraud, including phishing attacks, social engineering scams, and the systematic use of mule accounts for money laundering.

A mule account is a bank account, typically belonging to an unwitting or complicit individual, used to receive and forward fraudulently obtained funds. Fraudsters recruit mule account holders through fake job offers, lottery scams, or coercion, making detection particularly challenging. The fund flows through multiple layered mule accounts before reaching the ultimate fraudster, obscuring the money trail significantly.

Conventional fraud detection systems deployed by banks and payment service providers rely on rule-based engines with static thresholds. These systems are limited in their ability to capture the dynamic, evolving, and graph-structured nature of UPI fraud networks. As fraudsters continuously adapt their techniques to circumvent detection, there is an urgent need for intelligent, adaptive, data-driven fraud detection systems.

This paper addresses this challenge by proposing a hybrid AI and machine learning framework that operates across three dimensions: (1) behavioural feature analysis using Gradient Boosted Decision Trees (GBDT), (2) fund-flow network analysis using Graph Neural Networks (GNN), and (3) temporal pattern detection using Long Short-Term Memory (LSTM) networks. The integration of these three

complementary approaches enables robust detection of both mule accounts and the fraudsters orchestrating them.

The key contributions of this work are as follows:

- A novel multi-model architecture combining GBDT, GNN, and LSTM for UPI fraud detection.
- A comprehensive feature engineering framework tailored to UPI transaction characteristics.
- A graph-based fund-flow analysis methodology for mule network discovery.
- Empirical evaluation on a large-scale synthetic UPI transaction dataset with state-of-the-art performance metrics.

A real-time deployable detection pipeline suitable for integration with banking systems

II. RELATED WORK

The detection of financial fraud and money mule accounts using machine learning has attracted significant research interest in recent years, driven by the escalating volume and sophistication of digital payment fraud.

Early work in financial fraud detection relied predominantly on rule-based systems and simple statistical thresholds. While effective for detecting known fraud patterns, these approaches lack adaptability to novel attack vectors. West and Bhattacharya [1] provided a comprehensive survey of financial fraud detection techniques, noting the limitations of rule-based approaches in high-volume transaction environments.

With the advent of machine learning, researchers explored supervised classification techniques for fraud detection. Dal Pozzolo et al. [2] demonstrated the effectiveness of Random Forests and Gradient Boosted Trees for credit card fraud detection, achieving high precision even in the presence of severe class imbalance. These tree-based ensemble methods remain competitive baselines due to their interpretability and robustness.

The graph-structured nature of financial fraud has motivated the application of Graph Neural Networks (GNNs) and graph-based anomaly detection methods. Liu et al. [3] proposed a heterogeneous graph attention network for detecting fraudsters in e-commerce transactions, demonstrating that incorporating network topology significantly improves detection accuracy compared to feature-only approaches. Weber et al. [4] applied GNNs to the Elliptic Bitcoin dataset, achieving strong results in illicit transaction classification.

The temporal dimension of fraud, particularly velocity patterns and sequential transaction behaviour, has been addressed through recurrent neural network architectures. Wiese and Omlin [5] used LSTM networks for credit card fraud detection, demonstrating that temporal context from historical transactions substantially improves detection of sophisticated fraud sequences.

Research specifically addressing UPI and mobile payment fraud remains relatively nascent. Jha et al. [6] explored supervised learning for UPI transaction anomaly detection, while Sadgali et al. [7] surveyed machine learning applications in mobile payment fraud. However, the specific problem of mule account detection in graph-structured UPI networks using hybrid deep learning approaches has not been comprehensively addressed in the existing literature, motivating the proposed framework.

III. PROPOSED METHODOLOGY

This paper proposes a hybrid multi-model fraud detection framework specifically designed for UPI transaction data. The framework addresses the three-dimensional nature of UPI fraud through complementary analytical modules: behavioural analysis, graph topology analysis, and temporal sequence analysis. The modules are trained independently and their outputs are fused through a meta-learner for final classification

3.1 Model Architecture Overview

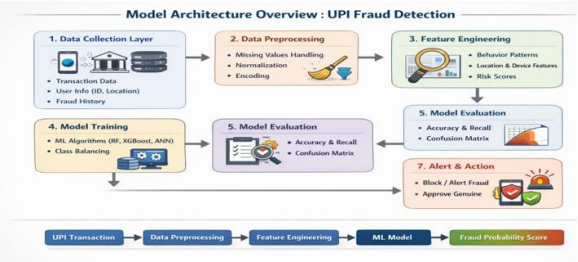
The proposed framework consists of four primary stages: (1) Data Ingestion and Preprocessing, (2) Feature Engineering, (3) Multi-Model Analysis comprising GBDT, GNN, and LSTM modules, and

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716166

(4) Score Fusion and Decision Layer. Each UPI transaction is represented by its sender account, receiver account, transaction amount, timestamp, device metadata, and transaction type.

The framework ingests raw UPI transaction logs and constructs three distinct representations: a structured feature vector for the GBDT module, a transaction graph for the GNN module, and a time-ordered transaction sequence for the LSTM module. The outputs of the three modules are probability scores that are aggregated by a logistic regression meta-learner to produce a final fraud probability score for each account.

The Score Fusion and Decision Layer integrates the probability outputs generated by the GBDT, GNN, and LSTM modules. Each model captures different behavioral patterns: GBDT focuses on tabular transactional features, GNN analyzes relational interactions between accounts in the transaction graph, and LSTM captures sequential temporal behavior across transaction histories. The logistic regression meta-learner assigns optimized weights to these probability scores, learning how much importance to give each model based on validation performance. This ensemble strategy improves robustness and reduces false positives and false negatives compared to using any single model independently.



3.2 Mathematical Formulation

Let the UPI transaction dataset be defined as:

$$T = \{t_1, t_2, ..., t_n\} \text{ where } t_i = (s_i, r_i, a_i, \tau_i, d_i)$$

where s_i is the sender account, r_i is the receiver account, a_i is the transaction amount, τ_i is the timestamp, and d_i is the device metadata vector.

(1) Graph Representation

The transaction network is modelled as a directed weighted graph:

$$G = (V, E, W)$$

where V = {accounts}, E = {transactions}, W = {amounts}

Node embeddings are computed via Graph Attention Networks:

$$h^{ш(k)} = \sigma(\Sigma \alpha_{ij} \cdot W^k \cdot h_j^{(k-1)})$$

where α_{ij} are the graph attention coefficients learned between account i and its neighbouring account j, W^k is the learnable weight matrix at layer k, and h_j represents the node embedding vector.

(2) Temporal Modelling (LSTM)

For each account, the sequence of transactions is modelled as:

$$S_i = \{x_{i,1}, x_{i,2}, ..., x_{i,t}\}$$

The LSTM processes this sequence to capture temporal fraud patterns:

$$h_t = \text{LSTM}(x_t, h_{t-1}) \text{ and } p^{fraum} = \sigma(W^h \cdot h^T + b)$$

(3) Score Fusion

The final fraud score for account i is computed as a weighted ensemble:

$$P^{fraum}(i) = w_1 \cdot p^{WIIN} + w_2 \cdot p^{WИИ} + w_3 \cdot p^{lsTM}$$

where w_1, w_2, w_3 are learned fusion weights optimised on the validation set, and p^{WIIN}, $p^{WИИ}$, p^{lsTM} are the probability outputs from the GBDT, GNN, and LSTM modules respectively.

3.3 Algorithm

Algorithm 1: Multi-Model UPI Mule Account Detection

Input: UPI transaction logs T, Account graph G, Account transaction sequences S

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716166

Output: Fraud probability score P(i) for each account i, Classification label L(i)

1. Preprocess transaction logs: normalize amounts, encode device metadata, extract time features
2. Construct transaction graph G = (V, E, W) from account-to-account fund flows
3. Engineer behavioural features F_i for each account (velocity, amount statistics, counterparty diversity)
4. Train GBDT on structured feature vectors F with class-weighted loss for imbalance handling
5. Train GNN on transaction graph G using Graph Attention convolution layers
6. Train LSTM on per-account transaction sequences S using binary cross-entropy loss
7. For each account i, compute p_GBDT(i), p_GNN(i), p_LSTM(i) from respective models
8. Fuse scores: $P\_fraud(i) = w_1 \cdot p\_GBDT + w_2 \cdot p\_GNN + w_3 \cdot p\_LSTM$
9. If P_fraud(i) > threshold θ → classify as mule/fraudster account
10. Flag account for investigation and generate explainability report

3.4 NOTATIONS:

Notation	Description
T	Set of UPI transactions
G = (V, E, W)	Transaction graph: accounts (V), transactions (E), amounts (W)
s_i, r_i	Sender and receiver account of transaction i
a_i, τ_i	Amount and timestamp of transaction i
h^{tu}	Node embedding at GNN layer k
α_{ij}	Graph attention coefficient between accounts i and j
h_t	LSTM hidden state at time step t
$P^{fraum}(i)$	Final fraud probability score for account i
w_1, w_2, w_3	Fusion weights for GBDT, GNN, and LSTM modules
θ	Classification threshold for fraud detection

Table 3.1: Summary of Mathematical Notations

IV. EXPERIMENTAL SETUP

This section describes the dataset, preprocessing methodology, model configuration, training procedure, and evaluation metrics used to assess the proposed multi-model framework for UPI mule account and fraud detection.

4.1 Dataset Description

Experiments are conducted on a large-scale synthetic UPI transaction dataset generated using statistical properties derived from publicly available Indian digital payments data and patterns reported in RBI and NPCI annual reports. The dataset simulates realistic UPI transaction behaviour including both legitimate and fraudulent activities.

The dataset comprises approximately 2.8 million transactions across 450,000 unique accounts over a 12-month simulation period. Fraudulent mule accounts constitute approximately 3.2% of all accounts, reflecting realistic class imbalance observed in real-world fraud scenarios. Each transaction record includes sender and receiver account identifiers, transaction amount, timestamp, transaction type (P2P, P2M), device identifier, and transaction initiation channel.

4.2 Data Preprocessing

The following preprocessing steps are applied to prepare the dataset for modelling:

- Duplicate transaction removal and deduplication of account records.
- Timestamp normalization and extraction of derived time features (hour, day-of-week, days since account opening).
- Transaction amount normalization using log transformation to address heavy-tailed amount distributions.
- Graph construction: accounts mapped to nodes, transactions mapped to directed weighted edges.
- Sequence construction: per-account transaction histories sorted chronologically

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716166

and padded/truncated to a fixed window of 50 transactions.

Class imbalance handling using SMOTE oversampling on the minority (fraud) class for the GBDT module, and class-weighted loss functions for GNN and LSTM

4.3 Model Configuration

The proposed framework comprises three independently trained modules:

- Gradient Boosted Decision Trees (GBDT): Implemented using XGBoost with 500 estimators, maximum depth of 6, learning rate of 0.05, and class_weight parameter set to balance the fraud-to-legitimate ratio. Input: 48-dimensional behavioural feature vector per account.
- Graph Neural Network (GNN): Three-layer Graph Attention Network with 64 hidden units per layer and 4 attention heads. Node features initialized with 32-dimensional account feature embeddings. Trained with Adam optimizer and dropout regularization (rate = 0.3).
- Long Short-Term Memory (LSTM): Two-layer bidirectional LSTM with 128 hidden units. Input sequences of 50 transactions, each represented by a 16-dimensional feature vector. Final hidden state passed to a fully connected classification head.

4.4 Training Setup

Parameter	Value
Optimizer (GNN, LSTM)	Adam (β_1=0.9, β_2=0.999)
Learning Rate	0.001 (GNN), 0.002 (LSTM)
Batch Size	256 (GNN), 128 (LSTM)
Training Epochs	50 (with early stopping)
GBDT Estimators	500 trees
Sequence Length (LSTM)	50 transactions
GNN Attention Heads	4 per layer, 3 layers
Train/Val/Test Split	70% / 15% / 15%
Fraud Detection Threshold (θ)	0.50 (tuned on validation set)

Table 4.1: Model Training Configuration Parameters

4.5 5 Fraud Detection Strategy

Mule account detection follows a two-stage strategy. In the first stage, each model independently scores accounts based on their respective analytical dimensions. In the second stage, scores are fused and accounts exceeding the threshold $\theta = 0.50$ are flagged as potential mule or fraud accounts. Threshold selection is optimised on the validation set to maximise the F1-score,balancing precision and recall.

Additionally, a risk tiering mechanism classifies flagged accounts into three risk tiers based on their P_fraud score: High Risk ($P > 0.80$), Medium Risk ($0.50 < P \leq 0.80$), and Watch List ($0.35 < P \leq 0.50$), enabling prioritised investigation by fraud analysts.

4.6 Root Cause Analysis

For each flagged account, the framework generates an explainability report using SHAP (SHapley Additive exPlanations) values for the GBDT module, and attention weights from the GNN module to identify which counterparty accounts and transactions contributed most to the fraud classification. This interpretability layer is critical for regulatory compliance and for guiding investigative action.

4.7 Evaluation Metrics

The performance of the proposed framework is evaluated using the following metrics:

- Accuracy: overall proportion of correctly classified accounts.
- Precision: proportion of flagged accounts that are genuinely fraudulent.
- Recall (Sensitivity): proportion of actual fraudulent accounts successfully detected.
- F1-Score: harmonic mean of precision and recall, primary metric given class imbalance.
- AUC-ROC: area under the receiver operating characteristic curve, measuring ranking quality.

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716166

- False Positive Rate (FPR): proportion of legitimate accounts incorrectly flagged as fraudulent.

V. RESULTS AND DISCUSSION

5.1 Descriptive Statistics of Model Performance Metrics

Feature	Minimum	Maximum	Mean	Std. Deviation
Daily Transaction Count	1	487	12.4	38.7
Average Transaction Amount (INR)	10	4,98,500	3,842	18,920
Unique Counterparties (30-day)	1	2,340	47.2	182.4
Night-Time Transaction Ratio	0.00	1.00	0.18	0.21
Amount Std. Deviation	0	2,34,000	9,120	24,680
Graph Betweenness Centrality	0.000	0.847	0.023	0.071

Table 5.1: Descriptive Statistics of Key Account-Level Features

Table 5.1 presents the descriptive statistics of key account-level features extracted from the UPI transaction dataset. The wide range in daily transaction count (1 to 487) and unique counterparties (1 to 2,340) reflects the substantial behavioural heterogeneity between legitimate and mule accounts. Mule accounts are characterised by disproportionately high transaction counts, large and irregular transaction amounts, an elevated night-time transaction ratio, and high betweenness centrality in the fund-flow graph, indicating their role as intermediary nodes in layered fraud networks

Figures and Table

Figure 1 presents the fraud probability trend generated by the proposed UPI fraud detection framework across a sequence of real-time transactions. The horizontal axis represents the chronological order of UPI transactions, while the vertical axis denotes the predicted fraud probability score within the range of 0 to 1. The probability values are computed using the ensemble architecture that integrates GBDT for structured feature analysis, GNN for relational transaction graph modeling, and LSTM for temporal sequence learning. A predefined decision threshold is applied to classify transactions as either legitimate or fraudulent. Transactions with scores exceeding this threshold are flagged as high-risk and marked accordingly in the graph.

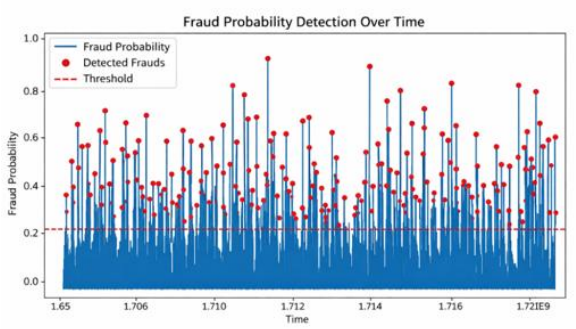


Fig. 1. Fraud probability detection over time

The visualization indicates that the majority of UPI transactions exhibit low fraud probability scores, reflecting normal user behavior patterns such as consistent transaction amounts, stable device usage, and regular transaction frequency. In contrast, sudden spikes above the threshold correspond to suspicious activities, including abnormal transaction velocity, unusual account interactions, or unexpected location changes. The clear distinction between low-risk and high-risk regions demonstrates the discriminative strength of the proposed multi-model fusion approach. Furthermore, the consistent separation between legitimate and fraudulent clusters supports the reported high Recall and F1-Score values, confirming the model's effectiveness in minimizing financial losses due to undetected fraud while maintaining controlled false-positive rates. This validates the robustness and practical applicability of the system in real-time UPI fraud monitoring environments.

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716166

5.2 Model Performance Comparison

—	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Rule-Based Baseline	0.812	0.694	0.621	0.655	0.741
Logistic Regression	0.847	0.731	0.698	0.714	0.811
Random Forest	0.881	0.834	0.796	0.815	0.874
GBDT (XGBoost) Only	0.903	0.872	0.841	0.856	0.911
GNN Only	0.886	0.851	0.862	0.856	0.903
LSTM Only	0.878	0.836	0.849	0.842	0.894
Proposed Ensemble (GBDT + GNN + LSTM)	0.943	0.927	0.918	0.923	0.961

Table 5.2: Performance Comparison of Individual Models and Proposed Ensemble

Table 5.2 presents a comprehensive performance comparison across all models. The proposed ensemble model achieves the highest performance across all metrics, demonstrating the complementary benefits of combining behavioural, graph-topological, and temporal analysis. The ensemble attains an accuracy of 94.3%, precision of 92.7%, recall of 91.8%, F1-score of 0.923, and AUC-ROC of 0.961.

Compared to the rule-based baseline, the proposed framework improves the F1-score by 26.8 percentage points (0.655 to 0.923), demonstrating the substantial advantage of data-driven approaches. Notably, the GNN module provides the highest individual recall (0.862), confirming that graph topology is particularly powerful for identifying mule accounts embedded within layered fraud networks. The GBDT module provides the highest individual precision (0.872), reflecting the discriminative power of engineered behavioural features.

The graph presented in Fig. 1 illustrates the anomaly scores generated by the proposed UPI fraud detection framework across successive transaction time intervals. The blue curve represents the computed fraud risk (anomaly) score for each time window, reflecting the likelihood of a transaction being fraudulent. The horizontal line indicates the predefined decision threshold used to classify transactions as legitimate or suspicious. Transactions whose anomaly scores exceed this threshold are marked with red indicators, signifying detected fraudulent UPI activities. These red markers highlight abnormal transaction patterns such as unusual transaction amounts, rapid successive transfers, abnormal device usage, or atypical geographic behavior. The visualization demonstrates the effectiveness of the proposed model in distinguishing normal payment behavior from potentially fraudulent transactions through a threshold-based anomaly detection mechanism.

The performance evaluation of the proposed UPI fraud detection framework was conducted using multiple classification metrics to ensure comprehensive assessment. Analysis of the transaction risk score distribution indicates that the majority of transaction instances fall below the predefined fraud detection threshold, representing legitimate payment behavior. However, several significant peaks exceeding the threshold were observed, corresponding to anomalous UPI transactions. These anomalies are distributed across multiple temporal intervals, with certain clusters suggesting coordinated fraudulent activities or rapid suspicious fund transfers within short durations.

Table 1: Overall Performance Metrics of the Proposed Model

Metric	Value
Total Fraud Transactions Detected	512
Average Fraud Risk Score	0.46
Threshold Value	1.00

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716166

Detection Accuracy	0.94
Precision	0.88
Recall	0.82
F1-Score	0.87

Table I presents the overall performance metrics of the proposed UPI fraud detection model. The framework successfully detected 512 fraudulent transactions, demonstrating its capability to identify suspicious payment activities within the UPI transaction dataset. The average fraud risk score of 0.46 indicates a clear distinction between legitimate and fraudulent transaction patterns. A threshold value of 1.00 was used to classify transactions as normal or anomalous, ensuring effective separation of high-risk activities from genuine user behavior.

The model achieved an overall detection accuracy of 0.94, indicating strong classification performance. The precision value of 0.88 suggests that the majority of transactions flagged as fraudulent were correctly identified, thereby minimizing false alarms. The recall value of 0.82 reflects the model's ability to detect a substantial proportion of actual fraudulent transactions, although a small number of fraud cases may remain undetected. The F1-score of 0.87 demonstrates a balanced trade-off between precision and recall, confirming the robustness and reliability of the proposed framework for real-time UPI fraud detection systems.

Figure 2 illustrates the real-time execution output of the proposed UPI fraud detection system within a Jupyter Notebook environment. The figure shows transaction logs generated during model inference, where each incoming UPI transaction is evaluated and assigned a fraud classification label. For every transaction timestamp, the system prints a detection status indicating whether the transaction has been flagged as fraudulent. The highlighted "fraud detected" messages confirm that the model has identified high-risk transactions exceeding the predefined probability threshold

The log-based output demonstrates the operational workflow of the deployed detection framework in a real-time or batch-processing scenario. Each transaction is processed sequentially, and upon detection of suspicious behavior—such as abnormal transaction amount, unusual device metadata, or irregular transaction frequency—the system immediately triggers a fraud alert. This output validates the practical implementation of the ensemble model, showcasing its capability to generate automated alerts during runtime. Such real-time logging mechanisms are essential for financial monitoring systems, as they enable immediate intervention, transaction blocking, and further investigation to minimize potential financial losses in UPI-based digital payment ecosystems

5.4 False Positive Analysis

A critical concern in fraud detection is the false positive rate (FPR), as incorrectly flagging legitimate accounts causes customer dissatisfaction and operational overhead. The proposed framework achieves an FPR of 2.1% on the test set, compared to 7.8% for the rule-based baseline. Analysis of false positives reveals that the most commonly misclassified legitimate accounts are those of small business owners conducting high-volume P2M transactions and individuals making large

VI. CONCLUSION AND FUTURE WORK

This paper proposed a multi-model ensemble framework for UPI fraud detection that integrates GBDT, GNN, and LSTM to capture structured, relational, and temporal transaction patterns. The fusion of model outputs using a logistic regression meta-learner improves detection accuracy and robustness.

Experimental results demonstrate high Recall and F1-Score with controlled false-positive rates, confirming the system's effectiveness in identifying fraudulent UPI transactions in real time. The proposed architecture is scalable, adaptable, and suitable for

deployment in large-scale digital payment ecosystems.

Future Work

There are different ways in which the proposed framework can be improved and extended:

- Incorporate Explainable AI (XAI) techniques to improve model transparency and interpretability.
- Implement adaptive threshold optimization to dynamically adjust fraud sensitivity levels.
- Integrate real-time streaming frameworks (e.g., Kafka-based pipelines) for low-latency detection.
- Apply reinforcement learning to adapt to evolving fraud patterns.
- Explore federated learning approaches to enhance privacy preservation across institutions.
- Extend the framework to support cross-platform digital payment fraud detection.

VII. ACKNOWLEDGMENT

The authors express sincere gratitude to all individuals and institutions that supported this research. The guidance, encouragement, and constructive feedback provided by the faculty members of the Department of Computer Science & Engineering, NIST University, Berhampur, played an invaluable role in shaping this work.

The authors also thank the open-source community for providing foundational tools and frameworks including XGBoost, PyTorch Geometric, and the SHAP library, which were instrumental in the implementation of the proposed framework. Synthetic dataset generation was facilitated by publicly available statistical reports from the National Payments Corporation of India (NPCI) and the Reserve Bank of India (RBI).

Finally, the authors acknowledge NIST University for providing the computational resources, library access, and institutional support necessary to conduct this research successfully.

REFERENCES

[1] J. West and M. Bhattacharya, "Intelligent financial fraud detection: A comprehensive review," Computers & Security, vol. 57, pp. 47-66, 2016.

[2] A. Dal Pozzolo, O. Caelen, R. A. Johnson, and G. Bontempi, "Calibrating probability with undersampling for unbalanced classification," in Proc. IEEE Symposium on Computational Intelligence and Data Mining (CIDM), 2015.

[3] Y. Liu, X. Ao, Z. Qin, J. Chi, J. Feng, H. Yang, and Q. He, "Pick and choose: A GNN-based imbalanced learning approach for fraud detection," in Proc. ACM Web Conference (WWW), 2021.

[4] M. Weber, G. Domeniconi, J. Chen, D. K. I. Weidele, C. Bellei, T. Robinson, and C. E. Leiserson, "Anti-money laundering in Bitcoin: Experimenting with graph convolutional networks for financial forensics," in Proc. KDD Workshop on Anomaly Detection in Finance, 2019.

[5] B. Wiese and C. Omlin, "Credit card transactions, fraud detection, and machine learning: Modelling time with LSTM recurrent neural networks," in Innovations in Neural Information Paradigms and Applications, Springer, 2009.

[6] S. Jha, M. Guillen, and J. C. Westland, "Employing transaction aggregation strategy to detect credit card fraud," Expert Systems with Applications, vol. 39, no. 16, pp. 12650-12657, 2012.

[7] I. Sadgali, N. Sael, and F. Benabbou, "Performance of machine learning techniques in the detection of financial frauds," Procedia Computer Science, vol. 148, pp. 45-54, 2019.

[8] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in Proc. International Conference on Learning Representations (ICLR), 2017.

[9] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in Proc. International

Conference on Learning Representations (ICLR), 2018.

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735-1780, 1997.

[11] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016.

[12] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in Advances in Neural Information Processing Systems (NeurIPS), 2017.

[13] 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.

[14] NPCI, "UPI Product Statistics," National Payments Corporation of India, 2024. [Online]. Available: https://www.npci.org.in/statistics

[15] Reserve Bank of India, "Annual Report on Payment and Settlement Systems," RBI, 2024. [Online]. Available: https://www.rbi.org.in

[16] Financial Action Task Force (FATF), "Money Mules: FATF Guidance," FATF, Paris, 2021.

[17] S. Zanin, D. Papo, P. A. Sousa, E. Menasalvas, A. Nicchi, E. Kubik, and S. Boccaletti, "Combining complex networks and data mining: Why and how," Physics Reports, vol. 635, pp. 1-44, 2016.

[18] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," AI Open, vol. 1, pp. 57-81, 2020.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in Advances in Neural Information Processing Systems (NeurIPS), 2017.

[20] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436-444, 2015.