International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 03 March 2026
Page No: 1160-1169
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0164

AI Based Transaction Anomaly and Fraud Detection System

Ms. Sana Mohammad Sadique Shaikh[1], *Mr. Goden N. A*[2]
[1,2] *V.V.P. Institute of Engineering & Technology, Solapur, India*
Email ID: *shaikhsana456@gmail.com*[1], *nageshgoden@gmail.com*[2]

Abstract

The fast growth of digital payment systems has led to a substantial rise in both the volume and complexity of fraudulent financial transactions. Traditional rule-based fraud detection methods lack flexibility and often fail to recognize new or previously unknown fraud patterns.This paper presents a hybrid explainable artificial intelligence (XAI) framework for real-time financial fraud detection. The proposed approach combines supervised learning using the XGBoost algorithm with unsupervised anomaly detection through Isolation Forest, enabling the system to identify both known fraud types and emerging suspicious activities.To enhance detection performance, the framework incorporates behavioral profiling, temporal transaction analysis, merchant risk evaluation, and device-level consistency features. Additionally, explainable AI techniques based on SHAP are applied to generate clear and interpretable explanations for each fraud prediction.Experimental results show that the hybrid model achieves higher recall and a lower false-positive rate compared to individual models, demonstrating its effectiveness and suitability for deployment in modern banking systems and digital payment platforms.

Keywords: *Financial FraudDetection, Anomaly Detection, XGBoost, IsolationForest, Explainable, AI, Digital Payments.*

1. Introduction

The increasing adoption of online banking, mobile wallets, and real-time payment platforms has transformed global financial systems. While these technologies offer speed and convenience, they also introduce significant security challenges. Financial fraud has become more sophisticated, involving identity theft, account takeovers, merchant impersonation, and transaction laundering, resulting in substantial financial and reputational losses for institutions and customers alike [1], [3]. Traditional fraud detection systems rely primarily on static rules and manually defined thresholds. Although these systems are interpretable, they lack adaptability and require continuous manual updates to remain effective [3], [5]. As fraud strategies evolve rapidly, such approaches often fail to detect complex and previously unseen attack patterns. Machine

learning (ML) methods offer a data-driven solution by automatically learning transaction behavior patterns from historical financial data [7], [11]. Supervised models such as decision trees, random forests, and gradient boosting have shown strong performance in fraud classification tasks [11], [19], [21]. However, these methods depend heavily on labeled data, which is typically scarce and highly imbalanced in fraud detection scenarios [20]. Unsupervised anomaly detection techniques, including Isolation Forest and autoencoders, address this limitation by identifying rare and abnormal transactions without requiring labeled fraud samples [22], [31]. Nevertheless, unsupervised models may generate a high number of false positives when used in isolation. To overcome these limitations, this research proposes a hybrid fraud detection framework that combines supervised and unsupervised learning with explainable AI. The objective is to achieve real-time detection, robust generalization to new fraud patterns, and transparent decision-making suitable for regulatory and operational requirements.

2. Literature Review

Fraud detection has been widely studied as an anomaly detection problem in highly imbalanced and evolving datasets [1]. Early statistical and rule-based approaches focused on identifying deviations from normal transaction behavior but suffered from limited adaptability and scalability [3].With the advancement of machine learning, supervised classification methods became dominant. Bhattacharyya et al. demonstrated that ensemble-based classifiers outperform single models due to their ability to capture nonlinear feature interactions [11]. XGBoost, in particular, has gained popularity in financial fraud robustness to noisy data [21].However, supervised approaches struggle with the rarity of fraud cases and concept drift, where transaction behavior changes over time [20]. To address this, unsupervised anomaly detection methods such as Isolation Forest have been proposed, which isolate rare observations efficiently in high-dimensional spaces [22].Comparative studies show that no single anomaly detection algorithm consistently outperforms others across all datasets [31].Recent research highlights the effectiveness of hybrid models that combine supervised and unsupervised techniques [13]. Such systems leverage labeled data to detect known fraud patterns while using anomaly detection to identify emerging threats. Additionally, explainable AI has become increasingly important in financial applications, as regulatory compliance and analyst trust require transparent decision-making [60]. Despite these advancements, challenges remain in achieving real-time detection, reducing false positives, and providing actionable explanations, motivating the proposed framework.

Table 1 **Methodology**

Method (Existing Work)	Technique Used	Limitation in Existing Paper	Our Improvement
Logistic Regression	Supervised linear classification	Fails to capture complex and non-linear fraud patterns	We use XGBoost to model complex feature interactions
Random Forest (Breiman, IEEE)	Ensemble decision trees	Requires large labeled datasets and struggles with evolving fraud patterns	Our hybrid model integrates anomaly detection for unseen fraud
XGBoost (Chen & Guestrin)	Gradient boosting trees	Detects only known fraud patterns from labeled data	We combine XGBoost with Isolation Forest to handle zero-day fraud
Isolation Forest (Liu et al.)	Unsupervised anomaly detection	Produces high false positives when used alone	A hybrid decision engine reduces false alarms using supervised scores
Hybrid Learning (Carcillo et al.)	Supervised and unsupervised fusion	Limited explainability and deployment focus	We add SHAP-based explainability and real-time deployment analysis
Rule-Based Systems	Static rule matching	Poor scalability and adaptability	Our system is scalable, adaptive, and capable of real-time processing

3. Search Gaps Research Gaps

Based on the literature, the following gaps are identified:

- Limited real-time detection capability, with many systems operating in batch mode [55]
- 2. Inability to detect zero-day fraud patterns due to reliance on historical labels [22]
- Lack of explainability in black-box ML models [60]
- Severe class imbalance, leading to biased learning and poor recall [20]
- Insufficient behavioral and contextual analysis, ignoring user habits and device consistency [11], [31].
- High false-positive rates, affecting customer

experience and operational cost [13]

- Limited adoption of hybrid and ensemble models in deployed systems

4. System Architecture

The system architecture represents the complete structural design of the proposed deep learning–based pneumonia detection framework. It illustrates how chest X-ray images are acquired, processed, analyzed using deep neural networks, interpreted through explainable AI techniques, and finally deployed for real-time clinical usage. The architecture is designed to ensure accuracy, robustness, interpretability, and ease of deployment in real healthcare environments.The proposed architecture consists of the following interconnected modules:

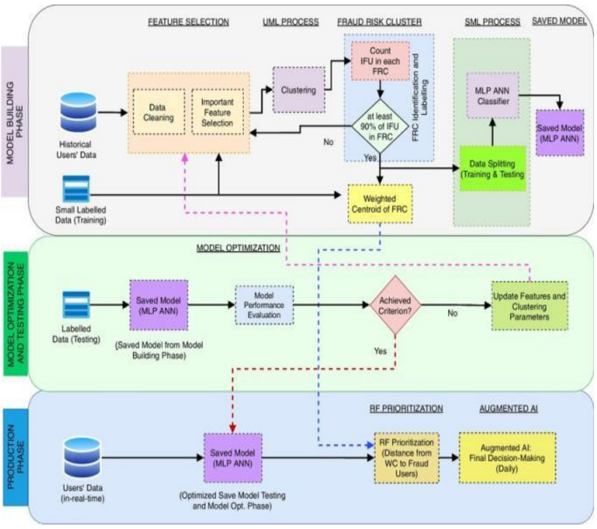
MODEL BUILDING PHASE
FEATURE SELECTION
UML PROCESS
FRAUD RISK CLUSTER
SML PROCESS
SAVED MODEL
Historical Users' Data
Data Cleaning
Important Feature Selection
Clustering
Count IFU in each FRC
at least 90% of IFU in FRC
FRC Identification and Labelling
No
Yes
MLP ANN Classifier
Saved Model (MLP ANN)
Data Splitting (Training & Testing
Small Labelled Data (Training)
Weighted Centroid of FRC
MODEL OPTIMIZATION AND TESTING PHASE
MODEL OPTIMIZATION
Labelled Data (Testing)
Saved Model (MLP ANN)
(Saved Model from Model Building Phase)
Model Performance Evaluation
Achieved Criterion?
No
Yes
Update Features and Clustering Parameters
PRODUCTION PHASE
RF PRIORITIZATION
AUGMENTED AI
Users' Data (in-real-time)
Saved Model (MLP ANN)
(Optimized Save Model Testing and Model Opt. Phase)
RF Prioritization (Distance from WC to Fraud Users)
Augmented AI: Final Decision-Making (Daily)

Figure 1 System Architecture

The proposed fraud detection framework is designed as a layered and modular architecture to ensure scalability, robustness, and real-time operational capability in modern digital payment environments. Each module performs a well-defined function while seamlessly interacting with other components to enable accurate and explainable fraud detection. The overall architecture is illustrated in Fig. X and is described in detail below.

4.1.Data Acquisition Layer

The Data Acquisition Layer is responsible for collecting raw transactional data from multiple financial channels, including online banking systems, UPI platforms, credit card networks, and e-commerce payment gateways. This layer ingests both historical and streaming transaction data, enabling real-time monitoring as well as offline model training.Each transaction record typically contains attributes such as transaction identifier, user identifier, transaction amount, timestamp, merchant category, payment method, device information, and geographical location. Collecting data from diverse sources allows the system to capture a comprehensive view of user behavior and transaction context, which is essential for effective fraud detection [7], [38].

This layer must support high-throughput data ingestion and low-latency processing, as delays in fraud detection can result in irreversible financial losses. In practical deployments, message queues or streaming platforms are often used to ensure reliable and scalable data flow [55].

4.2.Data Preprocessing and Feature Engineering Layer

Raw financial transaction data often contains missing values, noise, redundant attributes, and highly imbalanced class distributions. The Data Preprocessing and Feature Engineering Layer addresses these challenges to prepare high-quality input for machine learning models.Data cleaning operations include handling missing numerical values using statistical imputation and resolving incomplete categorical fields using default or inferred categories [1]. Numerical features such as transaction amount and inter-transaction time are normalized to ensure stable model training and convergence. Categorical attributes, including merchant category and device type, are encoded using suitable encoding techniques to make them compatible with ML algorithms [28].Feature engineering plays a critical role in fraud detection by transforming raw transaction data into meaningful behavioral indicators. Aggregated features such as transaction frequency within a time window, average spending deviation per user, location displacement between consecutive transactions, and device consistency metrics capture subtle fraud-related patterns that may not be visible in individual transactions [11], [15]. Prior studies demonstrate that behavior-based features significantly enhance fraud detection accuracy compared to using raw attributes alone [31].

4.3. Supervised Fraud Classification Module (XGBoost)

The Supervised Fraud Classification Module employs the XGBoost algorithm to classify transactions as fraudulent or legitimate based on labeled historical data. XGBoost is a gradient boosting framework that builds an ensemble of decision trees in a sequential manner, optimizing a loss function while controlling model complexity [21]. XGBoost is particularly well suited for financial fraud detection due to its ability to:

- Handle highly imbalanced datasets through weighted loss functions
- Capture nonlinear relationships between transaction features
- Provide feature importance scores for interpretability
- Scale efficiently to large transaction volumes

The supervised classifier learns known fraud patterns from past data and outputs a fraud probability score for each transaction. However, since labeled fraud data is limited and may not represent emerging attack strategies, supervised learning alone is insufficient for comprehensive fraud detection [20].

4.4.Unsupervised Anomaly Detection Module (Isolation Forest)

To detect previously unseen and evolving fraud patterns, the architecture integrates an Unsupervised Anomaly Detection Module based on Isolation Forest. Unlike supervised models, Isolation Forest does not rely on labeled fraud examples. Instead, it identifies anomalies by isolating rare and irregular data points through random feature partitioning [22]. In transaction fraud detection, anomalous transactions often differ significantly from normal user behavior in terms of amount, frequency, location, or device usage. Isolation Forest efficiently identifies such deviations even in high-dimensional feature spaces, making it suitable for real-time deployment [31]. The anomaly detection module generates an anomaly score for each transaction, representing its degree of deviation from normal behavior. This capability is crucial for identifying zero-day fraud attacks that have not been observed during model training.

4.5.Hybrid Decision Engine

The Hybrid Decision Engine is the core intelligence layer that combines outputs from the supervised classifier and the unsupervised anomaly detector. Instead of relying on a single model, the system uses a rule-based fusion strategy to improve robustness and reduce false negatives. A transaction is flagged as fraudulent if either the supervised fraud probability or the anomaly score exceeds predefined thresholds. These thresholds are optimized using validation data to balance detection sensitivity and false-positive rate [13]. This hybrid strategy allows the system to detect both known fraud patterns (via supervised learning) and novel anomalies (via unsupervised learning), addressing a major limitation of standalone models.Hybrid decision mechanisms have been shown to significantly improve recall while maintaining acceptable precision in real-world fraud detection systems [13], [20].

4.6.Explainability and Visualization Layer

Financial institutions require transparent and interpretable fraud detection decisions to comply with regulatory standards and maintain user trust. The Explainability and Visualization Layer addresses this requirement by integrating explainable AI (XAI) techniques. SHAP (SHapley Additive explanations) is used to quantify the contribution of each feature to a specific fraud prediction [60]. This allows analysts to understand why a transaction was flagged, such as unusual transaction amount, abnormal location change, or suspicious merchant behavior.In addition, a visualization dashboard presents fraud alerts, anomaly scores, feature importance plots, and performance metrics in an intuitive manner. This supports efficient investigation, auditability, and real-time monitoring of system performance.

4.7.Architectural Advantages

The modular design of the proposed architecture offers several advantages:

- Scalability: Each module can be independently scaled to handle high transaction volumes
- Flexibility: New models or features can be integrated without redesigning the entire system
- Real-Time Capability: Low-latency

processing enables instant fraud alerts

- Robustness: Hybrid intelligence improves detection of both known and unknown fraud patterns
- Transparency: Explainable outputs support regulatory compliance and analyst trust

These properties make the proposed architecture suitable for deployment in large-scale digital payment ecosystems.

5. Objective of Proposed System

- To develop an intelligent fraud detection system capable of identifying anomalous and fraudulent financial transactions in real time using machine learning techniques such as XGBoost, Isolation Forest, and Random Forest.
- To examine user transaction behavior by extracting important features including transaction amount, geographic location, frequency, merchant category, device information, and historical activity in order to detect unusual patterns.
- To enhance detection accuracy by effectively handling class imbalance through methods such as SMOTE, anomaly scoring, and optimized decision thresholds.
- To minimize false positives and false negatives by applying advanced hybrid and ensemble learning strategies.
- To design a scalable and adaptable fraud detection framework suitable for deployment across multiple financial platforms, including online banking, UPI services, credit card systems, and e-commerce applications.
- 6. To incorporate explainable artificial intelligence (XAI) techniques that provide clear and understandable reasons for each flagged transaction, thereby improving transparency and user trust.
- 7. To implement a secure processing pipeline that covers data preprocessing, model training, prediction, and alert generation while maintaining data confidentiality and system integrity.
- 8. To develop a user-friendly visualization dashboard (such as Streamlit) for presenting detectedanomalies, transaction patterns, risk scores, and model performance indicators.
- 9. To strengthen overall financial security by reducing the likelihood of unauthorized transactions through accurate and automated fraud detection.

6. Methodology

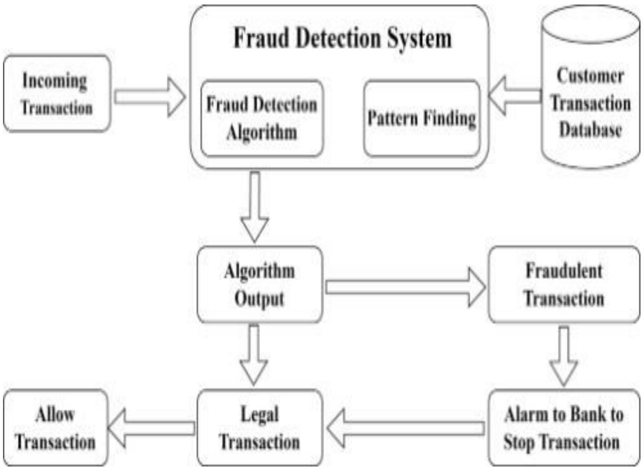

Figure 2 Workflow Diagram to Classify Fraud Transaction

6.1.Data Collection

The dataset consists of transactional records containing transaction identifiers, user IDs, transaction amount, merchant category, transaction type, timestamp, device information, geographical location, and fraud labels [38].

6.2.Data Preprocessing

Missing numerical values are imputed using median statistics, while categorical attributes are filled using mode or an “unknown” category [1]. Numerical features are normalized using standard scaling to improve model convergence. To address class imbalance, SMOTE and class-weight adjustments are applied [13], [20].

6.3.Feature Engineering

Derived features are designed to capture fraud-indicative behavior:

- Average transaction amount deviation per user
- Transaction frequency within sliding time windows

- Geographic distance between consecutive transactions
- Merchant risk profiling based on historical fraud rates
- Device usage consistency indicators

Behavioral aggregation has been shown to significantly improve fraud detection performance [11], [31].

6.4.Supervised Learning: XGBoost

XGBoost is employed to estimate fraud probability due to its ability to handle nonlinear interactions, imbalanced data, and large-scale datasets efficiently [21]. Hyperparameters are optimized using grid search to maximize recall while controlling false positives.

6.5.Unsupervised Learning: Isolation Fores

Isolation Forest identifies anomalies by recursively partitioning data points, effectively isolating rare and irregular transactions [22]. This component enhances the system’s ability to detect previously unseen fraud patterns.

6.6.Hybrid Decision Engine

A transaction is classified as fraudulent if:
Fraud probability $> \tau_1$ OR
Anomaly score $> \tau_2$
Thresholds are optimized using validation data
to balance recall and false-positive rate [13].

6.7.Concept Drift Detection

- Use ADWIN / DDM
- Monitor prediction error over time
- If drift detected:
- Trigger model retrainingEnsures adaptability to new fraud patterns

6.8.Explainable AI Integration

SHAP values are used to explain individual predictions by quantifying feature contributions [60]. This enables transparency, auditability, and trust in automated fraud detection decisions.

7. System Flow Chart

The system flowchart represents the end-to-end operational workflow of the proposed hybrid fraud detection framework. It illustrates how transaction data flows through different processing stages, from ingestion to final fraud decision and explanation. Each step is carefully designed to ensure accuracy, robustness, and real-time responsiveness.

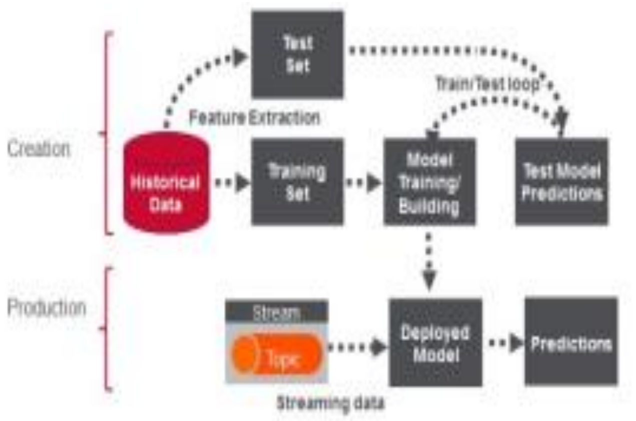

Figure 3 System Flowchart

7.1.Step 1: Transaction Initiation and Data Ingestion

The workflow begins when a financial transaction is initiated by a user through a digital payment channel such as online banking, UPI, credit card payment, or e-commerce checkout. Transaction details—including user ID, transaction amount, timestamp, merchant category, device information, and location—are captured and forwarded to the fraud detection system in real time.

This step ensures that every transaction is evaluated before final authorization, which is essential for preventing fraudulent fund transfers [7], [55].

7.2.Step 2: Data Validation and Preprocessing

Once the transaction data is received, it undergoes validation and preprocessing. This step includes:

- Checking for missing or inconsistent values
- Normalizing numerical attributes suchas transaction amount and time intervals
- Encoding categorical features such as merchant category and device type

Preprocessing ensures that incoming transaction data is consistent with the format used during model training, thereby preventing biased or unstable predictions [1], [28].

7.3.Step 3: Feature Engineering and Behavioral Profiling

In this stage, the system derives higher-level behavioral features from raw transaction data. These features capture user-specific and contextual transaction patterns, such as:

- Average spending deviation relative to historical behavior
- Transaction frequency within recent time windows
- Geographical distance between consecutive transactions
- Device usage consistency

Behavioral profiling allows the system to distinguish between legitimate unusual behavior and potentially fraudulent activity [11], [15]. This step is critical for reducing false positives.

7.4.Step 4: Supervised Fraud Probability Estimation (XGBoost)

The engineered feature vector is then passed to the supervised fraud classification model (XGBoost). The model computes a fraud probability score based on previously learned fraud patterns from labeled historical data. XGBoost effectively models nonlinear relationships between transaction features and fraud likelihood,aking it well suited for structured financial data [21]. This stage primarily detects known fraud patterns.

7.5.Step 5: Unsupervised Anomaly Scoring (Isolation Forest)

Simultaneously, the same transaction features are evaluated by the unsupervised anomaly detection model (Isolation Forest). This model assigns an anomaly score based on how isolated the transaction is compared to normal behavioral patterns. This step enables detection of previously unseen or emerging fraud patterns, which may not exist in historical labeled datasets [22], [31].

7.6.Step 6: Hybrid Decision Engine

The outputs from the supervised and unsupervised models are combined in the Hybrid Decision Engine. The decision logic evaluates whether:

- The fraud probability exceeds a predefined threshold, or
- The anomaly score exceeds an anomaly threshold

If either condition is satisfied, the transaction is flagged as potentially fraudulent. Thresholds are optimized using validation data to balance recall and false-positive rate [13], [20].

This hybrid fusion significantly improves detection robustness compared to single-model approaches.

7.7.Step 7: Explainable AI (XAI) Analysis

For every transaction flagged as suspicious, the system invokes the Explainable AI module. SHAP-based explanation techniques identify which features contributed most to the fraud decision, such as:

- Abnormally high transaction amount
- Sudden location change
- Unusual merchant category
- New or unrecognized device

Providing interpretable explanations improves analyst trust and supports regulatory compliance [60].

7.8.Step 8: Alert Generation and Visualization

If a transaction is classified as fraudulent, the systemgenerates a insights are showing: real-time alert. Alerts and transaction displayed on a monitoring dashboard

- Fraud probability and anomaly score
- Key contributing features
- Historical transaction comparison

This enables fraud analysts to take immediate action, such as blocking the transaction or initiating further verification.

7.9.Step 9: Transaction Approval or Rejection

Based on the hybrid decision and analyst or automated policy rules:

- Legitimate transactions are approved Suspicious transactions are blocked or sent for secondary verification.This final step ensures minimal disruption to genuine users while preventing financial loss [55].

7.10. Step 10: Feedback and Model Update (Optional)

Confirmed fraud and legitimate transaction outcomes are stored for future retraining. This feedback loop allows the system to adapt to evolving fraud strategies and improve detection performance over time [20].

Key Advantages of the Proposed Flowchart:

- Real-time fraud prevention before

transaction completion
- Dual intelligence using supervised and unsupervised models
- Reduced false positives through behavioral profiling
- Explainable decisions for audit and compliance
- Scalable and modular workflow suitable for large transaction volumes

8. Experimental Evaluation

8.1. Evaluation Metrics

The system is evaluated using accuracy, precision, recall, F1-score, ROC-AUC, and confusion matrices. Recall is emphasized due to the importance of minimizing undetected fraud [20].

8.2.Results and Discussion

Table 2 **Model and Precision**

Model **Precisio n (%)**	**Precisi on (%)**	**Recal l (%)**	**F1-Score (%)**	**ROC-AUC**
Logistic Regression	85.2	77.1	81.0	0.88
Random Forest	90.6	87.3	88.9	0.94
XGBoost	94.1	92.5	93.3	0.97
Isolation Forest	78.7	94.6	86.0	0.91
Proposed Hybrid Model	**93.5**	**96.3**	**94.9**	**0.98**

The hybrid framework achieves the highest recall while maintaining a low false-positive rate, validating its effectiveness for real-world deployment.

8.3.7.3 Scalability Analysis

The scalability of the proposed system is evaluated by increasing transaction volume from 1.2 million to 1.5 million users/transactions to simulate yearly growth. The framework maintains stable performance with only a small increase in processing time, showing that it can handle larger workloads efficiently and is suitable for real-time banking deployment.

8.4.Performance Analysis

The real-time performance is evaluated on different dataset sizes using execution time, response time, memory usage, and accuracy. The system processes transactions with low latency while maintaining high recall and precision. The average prediction time per transaction is suitable for real-world applications such as UPI, credit card, and e-commerce payments, ensuring minimal delay for genuine users and instant blocking of suspicious activities with very low real-time overhead.

Conclusion

This paper presented a hybrid explainable AI framework for real-time financial transaction fraud detection. By integrating supervised and unsupervised learning with explainable decision logic, the proposed system addresses key limitations of traditional fraud detection approaches. Experimental results demonstrate improved detection accuracy, robustness to emerging fraud patterns, and enhanced transparency, making the framework suitable for modern digital payment systems. The experimental evaluation confirms that the proposed framework introduces low computational overhead while supporting real-time fraud detection. Due to its modular architecture and efficient learning models, the system can be deployed in practical banking and digital payment environments with moderate hardware requirements. The scalability and performance results indicate that the framework is feasible for real-world deployment

Future Work

Future research directions include graph-based fraud detection using GNNs [54], continual learning to handle concept drift [20], blockchain-based audit trails, and federated learning for privacy-preserving fraud detection.The development of an AI-Based Transaction Anomaly and Fraud Detection System opens several promising avenues for future enhancement and real world deployment. As fraud patterns continue to evolve, integrating deep learning models such as LSTM, GRU, and Autoencoders can significantly improve detection accuracy and adaptability. Future systems may incorporate graph-based fraud analysis, allowing detection of fraud rings and complex multi account relationships. Expanding the dataset with real time and large-scale transactional data can further improve model robustness and minimize false positives. The integration of blockchain technology could strengthen data integrity and transparency while enabling secure audit trails. Additionally, embedding explainable AI (XAI) techniques will help financial institutions understand model decisions more clearly and comply with regulatory requirements. Future versions of the system can also provide mobile app integration, real-time push alerts, and role-based dashboards for enhanced usability. Ultimately, these advancements will support the creation of a highly scalable, intelligent, and comprehensive fraud detection ecosystem capable of securing digital transactions more effectively.

REFERENCES

[1]. Chandola, V., Banerjee, A., & Kumar, V. “Anomaly Detection: A Survey.” ACM Computing Surveys, 41(3), 1–58, 2009.

[2]. Ahmed, S., Mahmood, A. N., & Islam, M. R. “A Survey of Anomaly Detection Techniques in Financial Fraud.” Future Generation Computer Systems, 55, 278–294, 2016.

[3]. Bolton, R. J., & Hand, D. J. “Statistical Fraud Detection: A Review.” Statistical Science, 17(3), 235–255, 2002.

[4]. Phua, C., Lee, V., Smith, K., & Gayler, R. “A Comprehensive Survey of Data Mining-Based Fraud Detection Research.” arXiv:1009.6119, 2010.

[5]. Abdallah, A., Maarof, M. A., & Zainal, A. “Fraud Detection System: A Survey.” Journal of Network and Computer Applications, 68, 90–113, 2016.

[6]. Ravisankar, P., Ravi, V., Rao, G. R., & Bose, I. “Detection of Financial Statement Fraud.” Decision Support Systems, 50, 491–500, 2011.

[7]. Ngai, E. W., et al. “The Application of Data Mining Techniques in Financial Fraud Detection.” Decision Support Systems, 50, 559– 569, 2011.

[8]. Kou, Y., Lu, C. T., Sirwongwattana, S., & Huang, Y. P. “Survey of Fraud Detection Techniques.” IEEE International Conference on Networking, 2004.

[9]. Delamaire, L., Abdou, H., & Pointon, J. “Credit Card Fraud and Detection Techniques.” Banks and Bank Systems, 4(2), 57–68, 2009.

[10]. Fawcett, T., & Provost, F. “Adaptive Fraud Detection.” Data Mining and Knowledge Discovery, 1(3), 291–316, 1997.

[11]. Bhattacharyya, S., et al. “Data Mining for Credit Card Fraud.” Expert Systems with Applications, 2011. [12]Jha, S., Guillen, M., & Westland, J. “Employing Transaction Aggregation Strategy to Detect Fraud.” Expert Systems with Applications, 2012.

[12]. Carcillo, F., et al. “Combining Unsupervised and Supervised Learning in Fraud Detection.” Information Sciences, 557, 2021.

[13]. Maes, S., et al. “Credit Card Fraud Detection Using Bayesian Networks.” International NAISO Conference, 2002.

[14]. Whitrow, C., et al. “Transaction Aggregation for Credit Card Fraud Detection.” Pattern Recognition Letters, 2010.

[15]. Sahin, Y., & Duman, E. “Detecting Credit Card

[16]. Singh, A., & Jain, A. “Financial Fraud Detection Using Machine Learning.” IJCSE, 2020.

[17]. Brownlee, J. Machine Learning Algorithms in Python. Machine Learning Mastery, 2016.

[18]. [19] West, J., & Bhattacharya, M. “Detecting Payment Card Fraud.” Computers & Security, 2016.

[19]. Dal Pozzolo, A., et al. “Unbalanced DatasetsCredit Fraud Detection.”Big Data,2025.

[20]. Chen, T., & Guestrin, C. “XGBoost: A Scalable Tree Boosting System.” KDD Conference, 2016. Liu, F. T., Ting, K. M., & Zhou, Z.-H. “Isolation Forest.” ICDM, 2008.

[21]. Breiman, L. “Random Forests.” Machine

Learning, 2001.
[22]. Hochreiter, S., & Schmidhuber, J. “LSTM Networks.” Neural Computation, 1997.
[23]. Goodfellow, I., Bengio, Y., & Courville, A. Deep Learning. MIT Press, 2016.