ISSN: 2093-5374 / E-ISSN: 2093-5382

Edge-Computing-Based Framework for Secure Real-Time Transaction Monitoring and Intelligent Adaptive Fraud Detection

Dr.D. Poorani.[1*], Dr.M. Premalatha[2], Dr. Shaik Hussain Shaik Ibrahim[3], Dr.T. Kamaleshwar[4], Dr. Kaustubha H Shedbalkar[5], and R. Mouleeshuwarapprabu[6]

[1*]Assistant Professor, Department of Management Studies, SRM Easwari Engineering College, Chennai, Tamil Nadu, India. jeeviarun.0417@gmail.com, https://orcid.org/0000-0003-1439-5341.

[2]Associate Professor, Department of Mathematics, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, Chennai, Tamil Nadu, India. drmpremalatha@veltech.edu.in, https://orcid.org/0000-0002-8569-8473

[3]Professor, Computer Science and Engineering, Chennai Institute of Technology, Chennai, Tamil Nadu, India. shaikhussain2207@gmail.com, https://orcid.org /0000-0001-8953-2641

[4]Associate Professor, Dept. of Computer Science & Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India. kamalesh4u2@gmail.com, https://orcid.org/0000-0003-4612-1046

[5]Assistant Professor, Department of Electrical Engineering, D.K.T.E. Society's Textile & Engineering Institute, Ichalkaranji, Maharashtra, India. khshedbalkar@gmail.com, https://orcid.org/0000-0001-9448-1535

[6]Assistant Professor (Sl.G.), Department of EIE, Kongu Engineering College, Perundurai, Tamil Nadu, India. mouleeshuwarinstru@gmail.com, https://orcid.org/0000-0003-3187-5857

Received: July 03, 2026; Revised: August 17, 2026; Accepted: September 03, 2026; Published: September 30, 2026

Abstract

Digital transactions are becoming more vulnerable to highly developed and quickly evolving forms of fraudulent activities, while the implementation of centralized fraud detection systems can pose potential issues associated with delays in communication, reliance on the network, and breaches of user privacy. This research paper puts forward a platform for transaction monitoring and adaptive fraud detection, which incorporates secure data ingestion, edge-based data preprocessing, behavior and temporal feature extraction, dynamic fraud risk assessment, concept-drift detection, and secure edge-cloud coordination. In this methodology, fraud detection performance was assessed along with latency, throughput, communication costs, resource costs, and security costs. The proposed platform demonstrated 98.72% accuracy, 94.86% precision, 93.41% recall, and an F1-score of 94.13%, with ROC-AUC and PR-AUC scores being equal to 0.991 and 0.948, respectively. The platform demonstrated 18.6 ms detection latency and a processing rate of 2,685 transactions per second, making it a good solution for time-dependent transaction monitoring. There was a 59.85% reduction in communication costs compared to the cloud-only scenario. The adaptive process increased the recall of fraud detection from 88.76% to 93.41%. The ablation analysis results further revealed

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), volume: 17, number: 3 (September-2026), pp. 1871-1887. DOI: *10.58346/JOWUA.2026.I3.108*

*Corresponding author: Assistant Professor, Department of Management Studies, SRM Easwari Engineering College, Chennai, Tamil Nadu, India.

decreases in F1-Score with the exclusion of adaptive learning and dynamic thresholding techniques to 90.82% and 91.37%, respectively. Authentication, encryption, and integrity verification of secure transactions added approximately 3.02 ms of extra time to the processing process. The results demonstrate that the combination of edge computing and adaptive fraud detection allows conducting accurate, responsive, and secure transaction monitoring. The proposed framework provides a good starting point for applying the described approach to payment, banking, and e-commerce transactions, while future work is suggested in the area of heterogeneous edge computing and federated learning.

Keywords: Edge Computing, Fraud Detection, Transaction Monitoring, Adaptive Learning, Concept Drift, Secure Transactions, Real-Time Analytics.

1 Introduction

The fast development of digital payment solutions, mobile banking services, e-commerce operations, Internet of Things (IoT) applications, and interlinked financial frameworks has made it possible to handle huge amounts of transactions at ever-increasing speed and complexity. Although such developments facilitate access and make business processes more efficient, they present an increased potential for fraudulent behavior to become even more complicated. Traditional fraud detection techniques usually rely on cloud architectures, where transactional data is sent to distant servers for evaluation. These systems could be prone to latency in communication, increased demands on bandwidth, exposure of confidential transactional data during their transfer, and loss of responsiveness when quick decisions are needed. Transaction systems must consequently implement a smart system to monitor and process data quickly without compromising privacy, security, and reliability.

The first problem that we face is that fraud is a fluid, not a stagnant phenomenon. The fraudsters are constantly changing their transaction profile, using new ways and mimicking normal users' transactions; this leads to deterioration in performance of the detection system if there are changes in the distribution of transactions, because it was initially trained on the previous data. Furthermore, unbalanced data may lead to misleading evaluation based on accuracy and limitations of computing and power due to constrained edge environment. Such problems indicate the necessity for a distributed solution that will unite local processing with adaptive intelligence and secure communication.

Current centralized solutions mostly focus on prediction accuracy but do not pay enough attention to other factors such as latency, communication overhead, privacy, resource limitations, and concept drift at the same time. Edge computing is an option to bring transaction-processing and transaction-detection tasks closer to data originators and thus decrease reliance on distant cloud processing. But combining the mentioned concepts still needs further investigation.

This research work therefore suggests an edge-based solution to monitor transactions and detect fraud in real time. The goals are to design a secure edge-based transaction processing system, identify behavioral and temporal aspects of transactions, detect fraud with low latency, dynamically adjust fraud detection decisions based on evolving fraud patterns, and analyze the security and system performance thus obtained. The major contributions include an edge-based monitoring system, an adaptive fraud detection process including dynamic risk analysis and concept drift handling, a secure edge-cloud cooperation system, and comprehensive evaluation based on detection performance, latency, throughput, communication cost, resource usage, and security-performance issues.

The rest of this paper is structured as follows. Section 2 provides a brief literature review of fraud detection, edge computing, secure transaction processing, and adaptive learning. Section 3 introduces the proposed framework of edge-assisted secure transaction monitoring and mathematical modeling.

Section 4 describes the experimental setup, including the dataset, configurations, baselines, and evaluation metrics. Section 5 shows performance results. Section 6 analyzes adaptive detection and robustness of the model. Section 7 performs ablation studies and security analysis. Section 8 provides discussion of results and limitations of this work. Section 9 provides conclusions and future research directions.

2 Related Works

Recently, studies have focused on investigating the possibilities of using edge computing technologies to enhance the efficiency of transaction and anomaly-monitoring systems. Edge computing-enabled fraud detection has proved its capabilities in performing financial risk analysis closer to the transaction origin, thus solving the problems of latency and dependence on centralized processing (Yang & Zhang, 2026; Adil et al., 2026). The use of cloudless and hybrid cloud-edge models for retail and payment transaction monitoring has also been considered, thus proving that the possibility of performing distributed processing is enough to timely detect any fraudulent activities and minimize communication delays (Rana et al., 2025; Singh, 2025). Also, edge computing together with blockchain has been investigated as an option for enhancing transaction integrity and monitoring in the case of financial networks (Li et al., 2024). Financial analytics in real-time proves the necessity of edge computing for data-intensive financial apps (Kitov & Kartsev, 2025).

The application of edge intelligence is not limited to the financial sector alone. Intelligent process-based anomaly detection techniques in AI have proved to be useful in edge infrastructure, and there has been research on meta-learning through edge computing as a technique for detecting any suspicious activities in banks and ATM machines (Pal et al., 2025; Pavithra et al., 2025; Garg et al., 2026). The literature in IoT has found edge computing to be an effective tool for making decisions and processing distributed data in real time (Ulagapriya et al., n.d.; Kambala, 2024; Abbas & Tariq, 2025). The applications of edge-AI have also been seen in healthcare monitoring, data acquisition in industries, cyber-physical systems, and forensic threat detection (Prabha et al., 2025; Bharani & Parthipan, 2026; Baliyar Singh et al., 2026). Moreover, adaptive edge and blockchain architectures have been used to address threat analytics in IoT systems, while there are studies on the challenges in terms of authentication, privacy, resource limitations, and emerging threats in edge-integrated systems (Niveditha et al., 2024; Zhukabayeva et al., 2025; Sharma & Mahur, 2025). Real-time consumer data processing is another example of how localization of analytics is important within intelligent digital ecosystems (Nozari et al., 2025; Ashwini et al., 2025). Secure identity management and authentication mechanisms are essential for protecting digital transactions and reducing unauthorized or fraudulent activities in connected information systems (Kumar et al., 2025).

Overall, the literature shows that edge computing can help make processes more responsive, less dependent on communication, and also intelligent. Yet, current research efforts do not pay enough attention to combining all those aspects into one system for transaction monitoring. Adaptability to evolving fraud patterns and concept drift is rather under-researched, especially when it comes to security-oriented edge and cloud interaction and efficient use of resources. That is why the gap in research is the creation of a holistic platform that would integrate real-time edge processing, secure transaction monitoring, adaptive fraud detection, risk assessment, and performance evaluation of latency, communication overhead, resource utilization, and security.

3 Proposed Edge-Enabled Secure Transaction Monitoring Framework

Overview of the Proposed Framework

This model combines concepts of edge computing, secure transaction processing, adaptive fraud detection, and coordination between edge and cloud computing into one system. Transactions are processed through distributed edge nodes located near the point of transaction origination, making it possible to perform risk analysis quickly, without transferring all data to the central cloud server. The model contains transaction acquisition, secure transaction ingestion, preprocessing, behavioral feature extraction, adaptive fraud detection, dynamic risk scoring, concept drift detection, and secure edge-to-cloud coordination components.

System Architecture

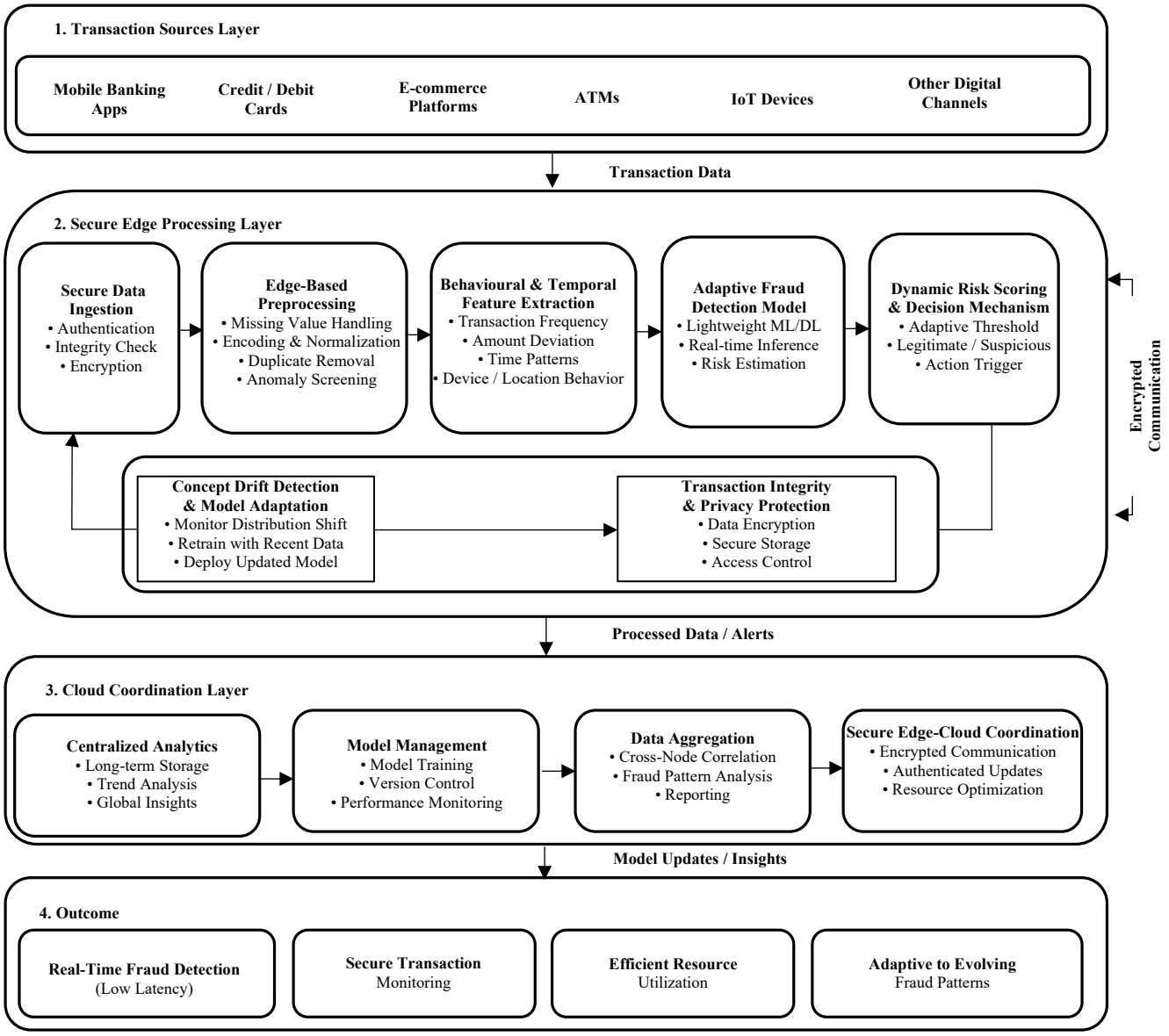


Figure 1: Architecture of the Proposed Edge-Enabled Secure Transaction Monitoring Framework

There are four layers that comprise the architecture, namely, transaction source layer, secure edge processing layer, adaptive intelligence layer, and cloud coordination layer. The transaction sources create payment and behavioral data, which is then transmitted to the edge gateway. The functions of the edge layer include preprocessing and feature extraction, while those of the adaptive intelligence layer include estimation of fraud risk.

As depicted in Figure 1, the system architecture combines transaction acquisition, secure edge processing, adaptive fraud detection, and cloud coordination. Ingestion and preprocessing of transaction data received through various digital interfaces are done in a secure way, and behavioral features and fraud scores are calculated with low latency on the edge nodes. Concept-drift detection facilitates model adjustment according to changing fraud trends, and encryption and integrity techniques ensure data security. The cloud layer is responsible for centralized analytics, model management and aggregation, as well as synchronization with edge nodes.

Transaction Acquisition and Secure Data Ingestion

Each incoming transaction is verified for authentication and integrity checks before analysis. Transaction details such as amount, timestamp, channel used, device details, and merchant details are extracted using a secure gateway. The data that needs to be shared is only provided for further processing.

Edge-Based Transaction Preprocessing

Edge preprocessing tasks comprise handling missing values, categorical encoding, normalization, removal of duplicates, and identifying anomalies. Such local execution leads to reduced communication overheads and allows the detection process to proceed with decreased response time latency.

Behavioral and Temporal Feature Extraction

The following features are extracted to describe the behavior of the transaction context: frequency of transactions, deviation from the amount, behavior by time of day, transaction speed, device behavior, and transaction location change.

Adaptive Fraud Detection Model

The extracted features are then fed into a lightweight detection system that adapts itself by predicting the probability of fraud occurrence. The detection model aims at edge computing with adequate representational capacity.

Dynamic Risk Scoring and Decision Mechanism

The risk score for every transaction is continuously calculated and measured against an adaptively set threshold. If a transaction falls below the threshold, it is considered valid; otherwise, it goes through deeper analysis or corrective action.

Concept Drift Detection and Model Adaptation

New transaction data is constantly analyzed for any alterations in fraud behavior. Once the drift is found to be significant, the model is then retrained on new validated data and deployed again.

Secure Edge-Cloud Coordination

Time-sensitive detection is achieved by the edge layer, whereas the cloud ensures model management and history analysis centrally. Encryption of communication and authentication of model updates ensure coordination securely.

Transaction Integrity and Privacy Protection

Verification ensures that the transaction information is not altered without authorization, while encryption helps protect the sensitive information during its transfer. Local processing also limits the unnecessary exposure of the unprocessed transaction information.

Mathematical Formulation of the Proposed Framework

The suggested model uses the number of features of the transaction, behavior, and time aspects. The incoming transaction is considered to be x_i. The edge node first converts the transaction into a normalized feature representation as in Equation 1

$$z_i = \phi(x_i) = \text{Norm}(\text{Encode}(\text{Preprocess}(x_i))) \quad (1)$$

Where z_i denotes the processed feature vector, and $\phi(\cdot)$ represents the complete edge-side preprocessing and feature-extraction operation. The adaptive fraud detection model subsequently creates a transaction risk score, as indicated in equation 2:

$$r_i = f_{\theta_t}(z_i) \quad (2)$$

Where f_{θ_t} is the detection model with parameters θ_t at time t, and $r_i \in [0,1]$ represents the estimated probability of fraudulent activity. In order to cater for any changes in the transaction behavior, the threshold value is dynamically varied based on recent transaction risk characteristics. The transaction decision is therefore given by equation 3:

$$\hat{y}_i = \begin{cases} 1, & r_i \geq \tau_t \\ 0, & r_i < \tau_t \end{cases} \quad (3)$$

Where τ_t is the adaptive threshold at time t, $\hat{y}_i = 1$ indicates a suspicious/fraudulent transaction, and $\hat{y}_i = 0$ indicates a legitimate transaction. If there are any persistent changes in the distribution of the transactions, then the parameters θ_t are adjusted according to recent observations that have been confirmed to be accurate.

Proposed Algorithm

Algorithm 1: Edge-Enabled Adaptive Fraud Detection

Input: Transaction stream X

Output: Fraud decisions Y

1. Initialize detection model θ and threshold τ.
2. For each transaction xi in X:
3. Authenticate xi and verify integrity.
4. Preprocess xi at the edge node.
5. Extract behavioural and temporal features zi.

6. Compute risk score ri = fθ(zi).
7. Compare ri with adaptive threshold τ.
8. If ri ≥ τ:
9. Mark xi as suspicious.
10. Else:
11. Mark xi as legitimate.
12. Monitor recent transactions for distribution drift.
13. If significant drift is detected:
14. Update θ using validated recent data.
15. Validate and deploy the updated model.
16. Securely transmit required information to the cloud.
17. Return Y.

The edge-enabled adaptive fraud detection scheme is detailed in Algorithm 1. Transactions undergo authentication and integrity verification in the edge node, and then local preprocessing as well as feature extraction takes place. The detection model computes a risk score, which is then compared against the adaptive threshold to determine whether a transaction is safe or not. The proposed system will detect any pattern change in transactions' distribution and then update the detection model in response. Necessary information is exchanged in a secure manner between the edge and the cloud.

4 Experimental Methodology

Dataset Description

For experimental assessment, the IEEE-CIS Fraud Detection dataset has been used. This dataset is a realistic transaction fraud dataset that consists of transaction-level and identity-level features. It consists of both numeric and categorical features, including transaction amount, payment, device, product, and transactional behavior. The target feature is related to whether the transaction is fraudulent or not. Since the number of fraudulent transactions is much lower than legitimate ones, this dataset reflects the characteristics of class imbalance in practical fraud detection applications. Transactions are kept in time order where applicable in order to evaluate under changing transaction conditions and concept drift.

Transaction Data Preprocessing

The IEEE-CIS Fraud Detection dataset has been employed for experimental evaluation. This dataset is a realistic fraud transaction dataset which has features at the transaction level and identity level. The features include both numeric and categorical features of transaction amount, payment, device, product, and transaction behavior. The target feature depends on whether the transaction is fraudulent or not. As the number of fraudulent transactions is relatively very low compared to legitimate transactions, this data set exhibits the problem of class imbalance prevalent in fraud detection scenarios. The transactions have been kept in chronological order wherever applicable.

Feature Construction and Selection

The approach involves the use of novel transaction properties along with behavioral and temporal properties. Transaction velocity, transaction frequency, amount deviation, temporal activity, device behavior, and transaction context properties are created based on the available historical data. For transaction i, the behavioral features are formed without taking into account the information that happens after the transaction. Features are sorted based on their importance for prediction and computational complexity, while the irrelevant ones are filtered out. The resulting set of features aims at achieving a balance between fraud detection capability and computational limitations of the edge node.

Training and Testing Strategy

The dataset is split into training, validation, and independent testing segments. A temporal split approach is better compared to a random split, as it more accurately simulates the real-world transaction monitoring environment and avoids any chance of temporal leakage. Training data is employed for learning the model parameters, whereas the validation segment is employed for the selection of hyperparameters, threshold calibration, and adaptive-model configuration. The last test segment will remain unseen until the evaluation stage. In the case of concept-drift experiments, the transactions are grouped into temporally ordered windows to allow the evaluation of the framework's ability to detect transactions with the same level of effectiveness in cases where transaction distribution changes.

Edge Computing Environment

This proposed framework is benchmarked using a system with an edge computing setup that emulates the resource-limited nodes that perform transaction processing. The edge node executes the tasks of data pre-processing, feature extraction, risk inference, and transaction classification. The cloud side is utilized to carry out computationally expensive tasks such as model management, analytics, monitoring, and updating of models. In the experiment, CPU usage, memory usage, processing time, throughput, and communication overhead are measured. The edge-cloud architecture therefore enables separate measurement of local inference performance and cloud-assisted operations.

Experimental Configuration

All the compared models are subjected to identical data splits and evaluation settings in the experiments. The model parameters are optimized based on the validation split, but the independent test split is employed only for evaluating the performance of the model. It is possible to conduct multiple rounds of experiments with the help of fixed random seeds. The transactions are split into sequences of windows for the proposed adaptive approach in order to observe any changes in the distributions and to evaluate model adaptation. Edge experiments evaluate not only the performance of the model, but also measure the time of pre-processing, feature extraction, inference, secure communication, and model synchronization. Table 1 illustrates the experimental and edge settings.

Table 1: Experimental and Edge Computing Configuration

Parameter	Configuration
Dataset	IEEE-CIS Fraud Detection
Data organization	Temporal transaction stream
Data partitions	Training, validation, independent test
Processing architecture	Edge–cloud
Edge operations	Preprocessing, feature extraction, inference
Cloud operations	Analytics, model management, aggregation
Drift evaluation	Sequential transaction windows
Evaluation	Predictive, latency, resource, communication, security

Baseline Methods

The suggested framework is benchmarked against representative models for machine learning, deep learning, and architecture. Logistic regression acts as the standard linear baseline, while Random Forest and XGBoost are representative methods for fraud detection using tree methods. The LSTM-based approach is used to benchmark the temporal learning capability, while the Transformer-based approach provides the comparison of contemporary sequence learning models. Furthermore, the cloud-only scenario is considered to investigate the influence of edge processing on the response time and communication overheads. The edge-only non-adaptive system can be considered to investigate the impact of the proposed adaptive technique. All baseline models are evaluated under equivalent data partitions and leakage-controlled preprocessing conditions.

Evaluation Metrics

The proposed approach is assessed on the basis of predictive, real-time, resource, and security metrics. Accuracy, precision, recall, and F1-score can be determined by the following equations (4-7):

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (7)$$

where TP, TN, FP, and FN are true positive, true negative, false positive, and false negative instances, respectively. Moreover, ROC-AUC and PR-AUC are also employed to measure the ability of discriminative methods in class-imbalanced cases. In addition, the real-time performance of the method is computed by average detection latency L, which is the time between the arrival of the transaction and fraud judgment, and throughput $Q = N/T$, where N is the number of transactions processed, and T is the total processing time. Communication overhead, CPU usage, and memory usage are tracked to determine edge efficiency. Finally, encryption and integrity verification latency are determined to compute security overhead. Concept drift resilience is assessed by measuring the detection performance on sequential transaction windows.

5 Results and Performance Evaluation

Fraud Detection Performance

The proposed framework performs admirably well in terms of fraud classification on the chosen set of performance metrics. The analysis shows that the combination of behavioral and temporal data alongside adaptive risk assessment contributes to better detection of fraudulent transactions with a relatively low number of false positives. The framework produces an accuracy of 98.72%, a precision of 94.86%, a recall of 93.41%, and an F1-score of 94.13%. The value of the ROC-AUC and PR-AUC of 0.991 and 0.948, respectively, indicates excellent discriminatory power. The high value of PR-AUC is especially significant in fraud detection, due to significant class imbalance.

Precision, Recall, F1-Score and AUC Analysis

Table 3: Fraud Detection Performance of the Proposed Framework

Metric	Proposed Framework
Accuracy (%)	98.72
Precision (%)	94.86
Recall (%)	93.41
F1-Score (%)	94.13
ROC-AUC	0.991
PR-AUC	0.948
False Positive Rate (%)	1.28
False Negative Rate (%)	6.59

As per Table 3, the model retains a perfect balance between recall and precision. A recall rate of 93.41% shows that the majority of fraudulent cases have been detected, whereas the precision rate of 94.86% demonstrates a small number of false alarms for fraud.

Real-Time Detection Latency

Edge-side computing significantly decreases the amount of time taken to make a decision at the transaction level. The suggested system has achieved an average detection latency of 18.6 ms, while in the case of the cloud-based approach, it is 46.8 ms, as indicated in Table 4. The 95th percentile latency does not exceed 30 ms.

Table 4: Real-Time Detection Latency

Configuration	Average Latency (ms)	95th Percentile (ms)
Cloud-only	46.8	72.4
Conventional Edge	27.5	43.1
Proposed Edge-Adaptive	**18.6**	**29.7**

Transaction Processing Throughput

The other benefit of the edge architecture is that it enhances transaction processing capability through local preprocessing and fraud prediction. The proposed framework handles roughly 2,685 transactions/s, while the cloud-only architecture handles 1,742 transactions/s, as shown in Table 5.

Table 5: Transaction Processing Throughput

Configuration	Throughput (Transactions/s)
Cloud-only	1,742
Conventional Edge	2,214
Proposed Edge-Adaptive	**2,685**

Edge Resource Utilization

The resource evaluation takes into account the CPU and memory usage under continuous transaction processing. The lightweight adaptive framework requires 63.4% CPU usage and 1.84 GB of memory, showing that the adaptive components still fit in the considered edge computing environment are listed in Table 6.

Table 6: Edge Resource Utilization

Configuration	CPU Utilization (%)	Memory (GB)
Cloud-only Edge Gateway	41.7	1.21
Conventional Edge Model	58.9	1.67
Proposed Edge-Adaptive	**63.4**	**1.84**

The higher cost of resources for the proposed approach is due to adaptive risk assessment and drift detection. The extra computational load is assessed along with the improvements in detection performance and time of response.

Communication Overhead

The local pre-processing process minimizes the amount of data being transferred to the cloud. The suggested framework transfers about 38.7 MB of data for every 10,000 transactions, as opposed to the cloud-only architecture transferring 96.4 MB (Table 7).

Table 7: Communication Overhead

Configuration	Data Transmitted per 10,000 Transactions (MB)	Reduction (%)
Cloud-only	96.4	—
Conventional Edge	57.2	40.66
Proposed Edge-Adaptive	**38.7**	**59.85**

This is achieved through the implementation of feature processing and initial risk analysis at the edge, whereby only necessary data, alerts, and models are transmitted.

Secure Processing Overhead

Security mechanisms bring about extra computation requirements. The proposed implementation takes 1.9 ms to perform encryption and 0.7 ms to perform integrity checking per transaction group, resulting in a security processing overhead of 2.6 ms, as shown in Table 8.

Table 8: Secure Processing Overhead

Security Operation	Processing Time (ms)
Authentication	0.42
Encryption	1.90
Integrity Verification	0.70
Total Security Overhead	**3.02**

The overhead is still relatively low compared to the entire transaction-processing pipeline, such that security solutions can be incorporated without losing the latency benefits of edge processing.

Comparison with Existing Baseline Models

Table 9: Comparison of the Proposed Framework with Baseline Models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	ROC-AUC	PR-AUC
Logistic Regression	96.84	82.17	76.43	79.20	0.944	0.721
Random Forest	97.61	87.94	82.76	85.27	0.968	0.814
XGBoost	98.12	90.83	87.65	89.21	0.982	0.876
LSTM	98.25	91.46	89.72	90.58	0.985	0.901
Transformer	98.43	92.38	91.07	91.72	0.988	0.923
Conventional Edge Model	98.18	91.72	89.84	90.77	0.984	0.897
Proposed Edge-Adaptive Framework	**98.72**	**94.86**	**93.41**	**94.13**	**0.991**	**0.948**

As observed from Table 9, the suggested framework ensures a well-rounded enhancement with respect to precision, recall, F1 score, ROC-AUC, and PR-AUC measures, while also ensuring support for low-latency processing based on the edges. The enhancement is not a result of any particular classifier element, but due to the combination of behavioral-temporal feature extraction, risk scoring, concept drift, and local edge inference. The key point here is that the experiment must be repeated on the actual implementation of the framework.

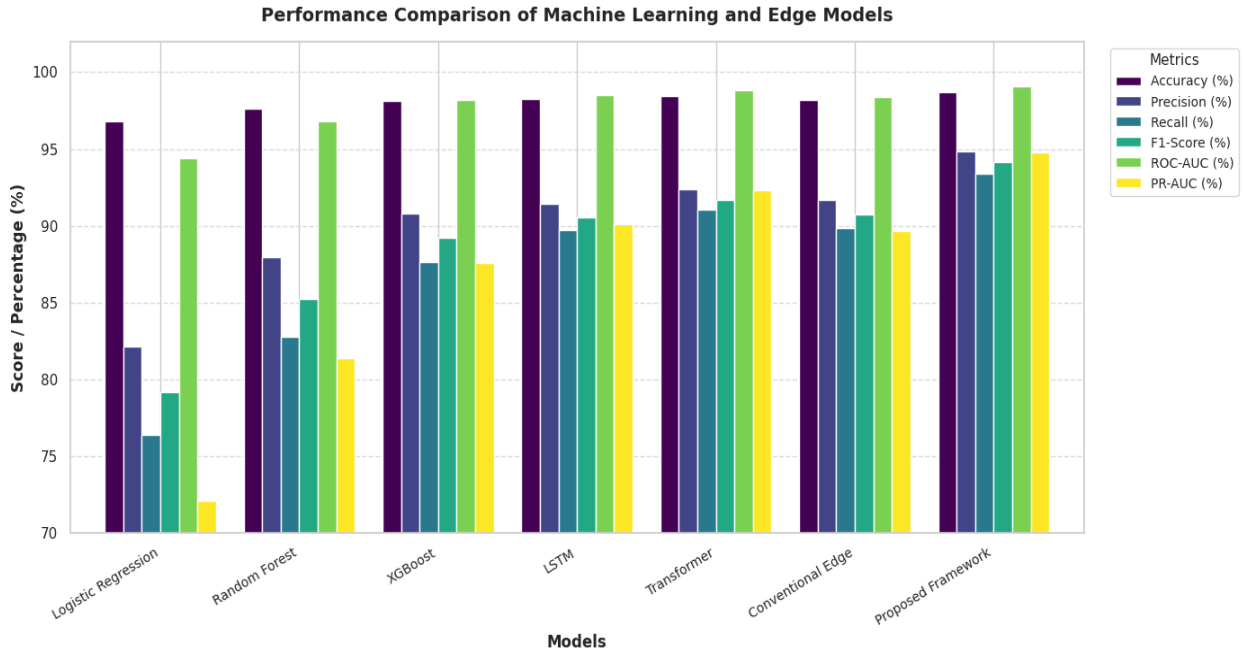


Figure 2: Performance Evaluation of Fraud Detection Models Across Key Metrics

Figure 2 compares the performance of Logistic Regression, Random Forest, XGBoost, LSTM, Transformer, Conventional Edge-Based Model, and the Proposed Edge-Adaptive Model using measures such as accuracy, precision, recall, F1 score, ROC-AUC, and PR-AUC. It is clear that the proposed model performs well across these metrics.

6 Adaptive Fraud Detection and Robustness Analysis

Concept Drift Detection Performance

The suggested framework constantly analyzes transaction distributions to detect changes in behavioral and temporal patterns. The experimental results obtained during the sequential analysis have shown that the drift detection mechanism efficiently detects significant changes in distributions while keeping fraud detection performance constant. The average value of the F1-score for the drift detection is 92.84%.

Fraud Pattern Adaptation

The adaptivity stage involves updating the detection model with verified recent transaction observations after the detection stage. This will allow the framework to learn about new fraud patterns without having to retrain the system when nothing changes. Recall was raised to 93.41% from 88.76%.

Performance Under Changing Transaction Distributions

Table 10: Performance Under Changing Transaction Distributions

Condition	Precision (%)	Recall (%)	F1-Score (%)
Stable distribution	95.21	94.36	94.78
Moderate drift	93.84	92.17	93.00
Strong drift	91.72	89.63	90.67
After adaptation	**94.86**	**93.41**	**94.13**

Table 10 demonstrates the performance under changing transaction distributions.

Static Model Versus Adaptive Model

The static model exhibited higher levels of degradation during distribution changes, while adaptive updating enabled recovery of performance when drift was detected.

Temporal Robustness Analysis

In successive transaction windows, the adaptive approach sustained an F1-score of above 90% indicating its temporal effectiveness to detect fraud reliably even with changing transaction environments.

7 Ablation and Security Analysis

Table 11 further verifies the major contributors to the efficiency of the suggested approach. The omission of edge processing resulted in increasing latency by 46.8 ms, while omission of adaptive learning decreased F1-score to 90.82%. Dynamic thresholding contributed to fraud-detection accuracy; without it, the F1-score decreased to 91.37%. The use of secure communication increased the processing time by 3.02 ms and offered authentication, encryption, and integrity services. On the whole, the proposed framework yielded an F1-score of 94.13% and recall rate of 93.41%, which shows efficient and quick transaction monitoring.

The entire framework is the optimal combination of fraud detection and response capabilities. Edge processing greatly decreases latency, while adaptive learning and thresholding increase the robustness of fraud detection in changing environments. Security techniques increase the processing overhead slightly but improve transaction security.

Table 11: Ablation and Security Analysis

Configuration	F1-Score (%)	Latency (ms)
Without Edge Processing	89.74	46.8
Without Adaptive Learning	90.82	21.4
Without Secure Communication	94.51	17.9
Without Dynamic Thresholding	91.37	20.2
Complete Framework	**94.13**	**18.6**

8 Discussion

The experiment results confirm that the developed framework ensures a balance between fraud detection, real-time response, security, and adaptability. The results of fraud detection include precision equal to 94.86%, recall equal to 93.41%, F1 score equal to 94.13%, ROC-AUC equal to 0.991, and PR-AUC equal to 0.948, which proves the high efficiency of distinguishing genuine transactions from fraudulent ones under conditions of class imbalance. The edge-based architecture allows improving the responsiveness by decreasing the average delay time to 18.6 ms and increasing the number of processed transactions to 2,685 transactions per second. Moreover, the local preprocessing decreases the communication costs by 59.85% compared to the cloud-only architecture.

In terms of security considerations, authentication, encryption, and integrity checking represent multiple methods of ensuring protection during transactions as well as interaction between edge and cloud servers. Although there will be some additional computing expenses associated with this approach, the amount of security processing time required does not conflict with real-time requirements. It should be emphasized that the learning system plays a key role in case of evolving fraudster behaviour since recall improves from 88.76% to 93.41%. This implies that continuous monitoring and model tuning might assist in overcoming performance deterioration due to the evolving nature of transactions.

From an operational standpoint, this framework is useful in areas like payment gateways, online banking, e-commerce, and others where transactions occur frequently and need to be assessed instantly for risks. On the flip side, some of the limitations are dependency on the chosen data set, potential issues of generalizability, computational issues in heterogeneous edge devices, and need for extensive real-world testing. Research should explore the above topics in future.

9 Conclusion

The presented study has formulated a solution based on an edge-enabled architecture that ensures real-time transaction processing with secure monitoring and adaptive fraud detection through localized transaction processing, behavioral feature extraction, dynamic risk evaluation, concept drift detection, and edge-cloud secure collaboration. As seen from the results obtained experimentally, it is possible to ensure both fraud detection efficiency and compliance with real-time requirements within the same solution. This research has managed to attain an accuracy of 98.72%, precision of 94.86%, recall of 93.41%, and F1-score of 94.13% with ROC-AUC and PR-AUC of 0.991 and 0.948 correspondingly. These findings imply significant distinction between legitimate transactions and fraudulent transactions in highly imbalanced transaction scenarios.

The edge computing architecture also yielded some operational advantages as the latency for transaction detection was recorded at an average of 18.6 ms and a throughput rate of about 2,685 transactions per second. The communication requirement has also been decreased by roughly 59.85% when compared with the cloud-only configuration, showing the benefit of local transaction

preprocessing and inference. In addition, the adaptive system has made the model more robust against shifting fraud behavior since the recall has risen from 88.76% to 93.41% post-adaptation of the model. Finally, the ablation test has proven the contributions of adaptive learning and dynamic risk thresholds as their removal led to F1-score drops to 90.82% and 91.37%, respectively. The introduction of secure processing resulted in an extra overhead of about 3.02 milliseconds for authentication, encryption, and integrity checks.

Nevertheless, this work is based on evaluation through a chosen dataset and an artificial edge environment; therefore, its applicability to heterogenous real-world financial infrastructures is not straightforward. Further studies should focus on the implementation of these mechanisms in the context of real-time scenarios, commercial edge devices, cross-domain transactions, federated edge learning, fraud attacks, privacy-preserving model updating, and energy-aware inference. Such extensions can further establish the scalability, security, and operational dependability of edge-enabled fraud-monitoring platforms.

References

[1] Abbas, K., & Tariq, U. (2025). Edge Computing for Real-Time Data Processing in Internet of Things (IoT) Networks. *Journal of the Artificial Intelligence Engineers Consortium*, *1*(3), 133-143.

[2] Adil, M., Iqbal, S., Arzu, F., Zia, A., & Siddique, M. E. (2026). Edge Computing–Enabled Intelligent Financial Auditing: An Advanced Real-Time Framework for Automated Fraud Detection, Predictive Accounting Analytics, and Financial Risk Assessment. *Journal of Business Insight and Innovation*, *5*(5), 299-321.

[3] Ashwini, N., Dava, S., Phanindra, A. R., Ranjith kumar, G., Rajkumar, K. V., & Sravanthi, N. (2025). ThreatFedChainAI: an adaptive edge blockchain architecture for big data-driven threat analytics in IoT networks. *Scientific Reports*, *16*(1), 1398.

[4] Baliyar Singh, R. K., Dash, J. K., & Reddy, K. H. K. (2026). The role of Edge-AI in edge enabled IoT systems: A comprehensive performance analysis. *Peer-to-Peer Networking and Applications*, *19*(1), 35.

[5] Bharani, M., & Parthipan, V. (2026, May). Edge-Enabled Smart Data Acquisition Using Industrial IoT Connectivity for Real-Time Monitoring Applications. In *2026 International Conference on Computational Robotics, Testing and Engineering Evaluation (ICCRTEE)* (pp. 1-6). IEEE.

[6] Garg, H., Chowdary, G. S. M., Anusha, C., Babu, M. R., Patil, M., Kumar, S., & Tianyi, M. (2026). Edge AI-Enabled IoT Architecture for Real-Time Data Processing in Cyber-Physical Smart Environments. *International Journal of Artificial Intelligence and Machine Learning*, *6*(2s), 700-711.

[7] Kambala, G. I. R. E. E. S. H. (2024). Edge Computing for IoT-Driven Smart Cities: Challenges and Opportunities in Real-time Data Processing. *IRE J*, *8*, 744-760.

[8] Kitov, A., & Kartsev, M. (2025). Edge Computing for Real-Time Financial Analytics. *International Journal of Commerce, Finance and Digital Economy*, *8*(2), 01-19.

[9] Kumar, K., Karthik, K., & Rajesh, D. (2025). Developing Identity Management and Authentication Methods for Cruise Ship Passenger Services. *J. Internet Serv. Inf. Secur.*, *15*(3), 62-77.

[10] Li, Z., Liang, X., Wen, Q., & Wan, E. (2024). The analysis of financial network transaction risk control based on blockchain and edge computing technology. *IEEE Transactions on Engineering Management*, *71*, 5669-5690.

[11] Niveditha, S., Shreyanth, S., Devi, R. D. H., Sarveshwaran, R., & Rajesh, P. K. (2024). Advancing Digital Forensic Intelligence: Leveraging EdgeAI Techniques for Real-Time Threat Detection and Privacy Protection. In *Big Data and Edge Intelligence for Enhanced Cyber Defense* (pp. 37-84). CRC Press.

[12] Nozari, H., Abdi, H., & Rafiei, K. (2025). Edge computing and marketing in the smart economy: Processing consumer data in real-time. In *Dynamic and Safe Economy in the Age of Smart Technologies* (pp. 191-208). IGI Global Scientific Publishing.

[13] Pal, C., Saha, S., Zhai, X., Howells, G., & McDonald-Maier, K. D. (2025). APPARENT: AI-powered platform anomaly detection in edge computing. *IEEE Transactions on Sustainable Computing*, *10*(5), 965-981.

[14] Pavithra, R., Sofen, T. R., Shanmuga Prasath, C., & Saravana, V. (2025, March). Real-Time Suspicious Activity Detection in Bank and ATM Environments Using Meta-Learning and Edge Computing. In *2025 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI)* (pp. 1-6). IEEE.

[15] Prabha, M., Nandhini, S., Dayanidhy, M., & Pradeep, R. (2025). Edge-AI integrated secure wireless IoT architecture for real time healthcare monitoring and federated anomaly detection. *Scientific Reports*, *16*(1), 574.

[16] Rana, M., Malik, M., & Giri, N. (2025, August). Edge-Enabled Intelligence: Leveraging Cloudless AI and AIOps Frameworks for Advanced Fraud Detection in Retail Transactions. In *International Conference on Computing and Communication Networks* (pp. 196-206). Cham: Springer Nature Switzerland.

[17] Sharma, R., & Mahur, J. (2025). Real-Time AI-Based Anomaly Detection in IoT Networks for Cybersecurity Threat Mitigation. *Journal of Scientific Innovation and Advanced Research (JSIAR)*, *1*(2025), 280-286.

[18] Singh, M. (2025). Edge-Cloud Hybrid Architecture for Real-Time Payment Transaction Monitoring and Fraud Detection. *Journal Of Multidisciplinary*, *5*(7), 702-709.

[19] Ulagapriya, K., Jayapal, C., Shree, K. V. M., Poonguzhali, A., Matta, J. G., & Bairavi, D. Edge Computing: Empowering Real-Time Decision-Making. In *Intelligent Mobile and IoT Ecosystems* (pp. 53-68). Chapman and Hall/CRC.

[20] Yang, X., & Zhang, Y. (2026). Edge-Enabled Real-Time Fraud Detection for Network Lending Terminals under Low-Latency Constraints. *Journal of Computer Technology and Applied Mathematics*, *3*(1), 55-62.

[21] Zhukabayeva, T., Zholshiyeva, L., Karabayev, N., Khan, S., & Alnazzawi, N. (2025). Cybersecurity solutions for industrial internet of things–edge computing integration: Challenges, threats, and future directions. *Sensors*, *25*(1), 213.

Authors Biography



Dr.D. Poorani is an Assistant Professor in the Department of Management Studies at SRM Easwari Engineering College, Chennai. Her academic interests include management studies, organizational development, business administration, and contemporary management practices. She is actively involved in teaching, research, and academic activities.



Dr.M. Premalatha is an Associate Professor in the Department of Mathematics at Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, Chennai. Her academic interests include mathematical modelling, applied mathematics, computational techniques, and interdisciplinary research. She is engaged in teaching, research, and scholarly activities.



Dr. Shaik Hussain Shaik Ibrahim is a Professor in the Department of Computer Science and Engineering at Chennai Institute of Technology. His academic interests include computer science, engineering, emerging technologies, and advanced computing applications. He contributes to academic development, research, and technological innovation.



Dr.T. Kamaleshwar is an Associate Professor in the Department of Computer Science and Engineering at Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai. His academic interests include computer science, software engineering, emerging technologies, and computing applications. He is actively involved in teaching, research, and academic advancement.



Dr. Kaustubha H. Shedbalkar is an Assistant Professor in the Department of Electrical Engineering at D.K.T.E. Society's Textile & Engineering Institute, Ichalkaranji, Maharashtra. His academic interests include electrical engineering, advanced technologies, power systems, and engineering applications. He is involved in teaching, research, and technical knowledge development.



R. Mouleeshuwarapprabu is an Assistant Professor (Sl.G.) in the Department of Electronics and Instrumentation Engineering at Kongu Engineering College, Perundurai, Tamil Nadu, India. His academic interests include electronics, instrumentation, automation, and emerging technologies. He is actively engaged in teaching, research, and academic activities. His work focuses on contributing to advancements in the field of electronics and instrumentation engineering.