Future Technology
Open Access Journal
ISSN 2832-0379

November 2026| Volume 05 | Issue 04 | Pages 108-118

Journal homepage: https://fupubco.com/futech

https://doi.org/10.55670/fpll.futech.5.4.9

Future Publishing LLC

Article

Leakage-aware federated-style simulation for fraud risk prediction in AI-enabled FinTech platforms

Jyoti Agarwal*, Moin Uddin

Department of Finance, College of Administrative and Financial Sciences, Saudi Electronic University, Riyadh, Saudi Arabia

ARTICLE INFO

Article history:
Received 16 February 2026
Received in revised form
24 June 2026
Accepted 25 July 2026

Keywords:
Federated-style simulation, Fraud-risk prediction,
Global feature importance,
Class-imbalanced learning,
FinTech fraud detection,
Resource-aware digital infrastructure

*Corresponding author
Email address:
j.agarwal@seu.edu.sa

DOI: 10.55670/fpll.futech.5.4.9

ABSTRACT

AI-enabled FinTech platforms increasingly rely on automated fraud-risk prediction to secure high-volume digital payments, mobile-money transactions, and real-time financial services. However, fraud detection remains difficult because fraudulent transactions are rare, financial data are sensitive, and centralized machine-learning models may create privacy and data-governance risks. This study evaluates a leakage-aware and imbalance-aware federated-style simulation for fraud-risk prediction in AI-enabled FinTech platforms. The study does not implement FedAvg, cryptographic secure aggregation, differential privacy, or model-parameter exchange. Instead, the PaySim Synthetic Financial Dataset was partitioned into five simulated FinTech clients, local models were trained separately, and prediction-level aggregation was used to compare centralized and simulated decentralized learning settings. The experimental workflow included data-quality assessment, leakage-aware feature selection, feature engineering, class-imbalance analysis, centralized model development, threshold optimization, and federated-style model comparison. The results show that fraud cases represented only 0.0466% of all transactions and occurred only in CASH_OUT and TRANSFER operations. The best-balanced centralized model was the Improved Random Forest at a threshold of 0.33, achieving an F1-score of 0.3019. Federated Ensemble Aggregation achieved the highest PR-AUC of 0.2699 and a precision of 1.0000, but its recall remained low at 0.1429. The centralized Improved Random Forest retained the highest F1-score of 0.3019, indicating a stronger overall precision-recall balance. Global feature-importance analysis showed that transaction amount, log-transformed amount, temporal features, transaction type, and destination-account activity were the main fraud-risk indicators. These findings should be interpreted as evidence from simulated decentralized fraud-risk learning rather than a full secure federated learning implementation.

1. Introduction

The fast proliferation of AI-enabled FinTech systems has dramatically changed the landscape of digital payments, mobile banking, credit facilities, and real-time transaction processing. Contemporary financial infrastructures tend to rely heavily on automated decision-making tools for detecting anomalies in user behavior and predicting risk and fraudulent transactions. With the growth of transaction volumes, traditional rule-based fraud detection models lose their efficacy due to the dynamics of fraud patterns and their hiding nature in large-scale financial data streams. Current advances in machine learning-based fraud detection have demonstrated that machine learning can facilitate real-time fraud detection by analyzing sophisticated transaction behaviors [1]. Concurrently, FinTech infrastructure tends to become distributed and oriented towards services. AI-powered microservices have been widely adopted in financial systems for achieving high levels of performance, scalability, and modularity in transaction processing [2]. At the same time, secure web and mobile app frameworks have been actively developed to provide user-oriented and AI-powered financial services in environments where trust, usability, and security coexist [3]. All these trends imply that fraud risk prediction should be viewed beyond its machine-learning aspects and considered a component of the secure FinTech ecosystem. AI has become an integral part of intelligent banking and FinTech transformation. AI-driven features support customer profiling, risk assessment, credit

evaluation, fraud monitoring, and decision-making processes [4]. On the other hand, the growing use of AI in FinTech entails risks related to the vulnerability of AI models to adversarial attacks, model manipulation, data poisoning, and privacy leakage in high-value financial infrastructures [5]. Hence, the fraud detection models need to be accurate, secure, privacy-preserving, and explainable.

There are four primary challenges associated with fraud risk prediction in FinTech platforms. Firstly, fraud cases are relatively rare compared to legitimate transactions, thus, leading to a critical class-imbalance problem. Secondly, financial transaction data are sensitive and cannot be exchanged between banks, mobile-money services, wallets, and other FinTech entities. Thirdly, centralized fraud detection models might require aggregation of sensitive customer data in one place, thus, causing privacy, regulatory, and cybersecurity issues. Finally, many AI models are black boxes, so there is no guarantee that financial institutions, auditors, and regulators would be able to understand the rationale behind classification of transactions as fraud. From the standpoint of computer and information technology, this study considers fraud detection a privacy-aware decentralized fraud-risk simulation rather than a regular classification problem. The proposed workflow involves leakage-aware preprocessing, imbalance-aware learning, threshold optimization, explainability, and federated-style client partitioning to analyze fraud-risk models' behavior in decentralized environments. Thus, this study belongs to the areas of advanced applied AI, privacy-aware data engineering, and secure digital financial systems.

The integration of blockchain and AI has been suggested as one of the ways to improve financial transaction security and transparency [6]. Likewise, the use of deep learning in anti-money laundering and mobile transaction monitoring demonstrates the necessity of developing advanced data-driven approaches to fraud risk prediction [7]. However, many approaches either do not address the issue of decentralization or fail to ensure privacy, security, and explainability. Hence, there is a gap in developing a framework for leakage-aware and imbalance-aware fraud-risk workflow in decentralized FinTech infrastructures without raw-data sharing. This study focuses on fraud risk prediction with structured mobile-money transaction data. For experiments, the PaySim Synthetic Financial Dataset representing mobile-money transactions with binary fraud labels is used. The scope of the study covers data quality assessment, leakage-aware feature selection, feature engineering, imbalance handling, centralized fraud-risk model development, threshold tuning, and federated-style learning simulation. Fraud prediction is evaluated using such metrics as precision, recall, F1-score, ROC-AUC, PR-AUC, false-positive rate, and confusion matrix.

This study is confined to the use of a 500,000-record PaySim sample and the implementation of federated-style simulation rather than FedAvg protocol. Nonetheless, this study serves as an experimental foundation for the further development of privacy-preserving and explainable fraud-risk prediction in decentralized FinTech environments. The significance of this study is based on its attempt to unite fraud prediction, federated-style learning, explainability, and security into a unified FinTech framework. The proposed approach is valuable for minimizing the necessity of centralized raw-data processing, thus, enabling more scalable and resource-efficient digital financial infrastructure, which aligns with sustainable and secure transformation of data-intensive financial technologies. The significance of AI-driven risk management and fraud prevention lies in its potential to improve digital finance, economic resilience, and trust in financial systems [8]. Also, the development of AI-powered credit and risk assessment systems implies the growing role of machine learning in financial activities in underbanked environments [9]. Thus, an AI-powered and explainable fraud prediction framework can be useful for supporting operational fraud protection and overall FinTech trustworthiness. The contributions of this study include the analysis of fraud risk prediction under severe class imbalance and decentralized data conditions. Imbalance-aware learning, threshold tuning, federated-style simulation, and feature-importance-based explainability were used to support privacy-aware and interpretable fraud-risk analysis in FinTech.

In addition, this study enhances the area of explainable AI by identifying the key fraud predictors, including transaction amount, log-transformed transaction amount, temporal features, transaction type, and destination account activity. Explainability is crucial for ensuring transparency and accountability of models and gaining regulatory approval. With the growing adoption of AI and blockchain convergence in connected financial and IoT environments, explainable AI frameworks need to provide secure and privacy-aware intelligent models [10]. It is important to clarify that this study does not implement FedAvg, cryptographic secure aggregation, differential privacy, or parameter-level federated optimization. The federated component is implemented as a simulation in which PaySim records are divided among five simulated FinTech clients, local models are trained separately, and prediction-level aggregation is used for comparison. The primary aim of this research is to evaluate a leakage-aware and imbalance-aware federated-style simulation for fraud-risk prediction in AI-enabled FinTech platforms. The specific aims are:

- To conduct a detailed analysis of the PaySim transaction dataset, including assessing data quality, fraud prevalence, transaction pattern, and imbalance issues.
- To design features for fraud detection that are resistant to data leakage and consider behavioral aspects based on temporal, monetary, frequency, and receiver features.
- To assess imbalance-aware machine learning models via techniques like class weighting, random undersampling, threshold adjustment, and fraud-specific performance measures like precision, recall, F1-score, ROC-AUC, PR-AUC, and false positive rate.
- To simulate decentralized fraud-risk prediction by creating multiple FinTech client partitions, training local models, comparing centralized and federated-style approaches, and assessing global feature-importance-based explainability.

2. Literature review

Modern AI-powered financial systems have shifted their focus toward intelligent data processing, cloud computing, and automated risk assessment. Recent research into the collaborative approach to cloud and AI in financially sensitive enterprise environments proves that combining distributed computing, intelligent automation, and intelligent data processing can ensure security and scalability in modern FinTech [11]. In particular, in the domain of FinTech fraud detection, the challenge lies in being able to scale up intelligent algorithms without compromising on security and privacy. However, increased reliance on AI in financial systems leads to emerging cybersecurity threats. On one hand, AI can help with enhancing bank cybersecurity through

anomaly detection and automated threat monitoring. On the other hand, AI models can become vulnerable to adversarial attacks, data leaks, and unauthorized access [12]. Mobile financial services are especially prone to security vulnerabilities since they rely on applications, payment mechanisms, and distributed digital identities. Mobile application security frameworks based on intelligent monitoring have proven that intelligent monitoring can improve the resilience of mobile applications against cyber threats [13]. Many FinTech fraud cases are committed via mobile wallet applications, payment applications, and online transaction applications. Thus, mobile-focused AI security studies represent a good basis for designing fraud prediction models. However, it is not enough to pay attention to security in fraud prediction. It is also necessary to ensure ethical responsibility, explainability, and accountability of AI models applied to financial decision-making. Existing research on ethically responsible machine learning in FinTech proves that there is a need for transparency, bias mitigation, accountability, and regulatory compliance when AI models are used for financial decision-making [14]. In this regard, fraud prediction models should be explainable so that banks could understand why the model predicts a specific transaction as fraudulent.

There is also another field related to the literature review – AI-based data engineering. Financial workflows imply the use of structured data processing techniques, data pipelines, feature preparation, scalable machine learning, and risk analysis [15]. In the case of fraud detection, feature engineering is crucial since suspicious activities might be detected through the temporal pattern of transactions, amount of money involved in a transaction, frequency of transactions, pattern of transaction recipients, or unusual transaction types. Thus, leakage prevention, preprocessing, and feature engineering are important when designing fraud prediction models. The current study follows these guidelines since it involves the engineering of transaction features based on PaySim data with the exclusion of balance attributes. Another direction in the existing literature is associated with blockchain and AI. Integration of blockchain and AI into FinTech processes enables improved security, traceability, and transaction monitoring [16]. Similarly, blockchain-based cybersecurity for cyber-resilient financial management frameworks relies on blockchain, AI, cloud systems, and auditing [17]. These studies prove that fraud detection models should be designed in a way that ensures cybersecurity and traceability. Nevertheless, many blockchain and cloud computing frameworks focus on providing concepts of cybersecurity and financial security architecture and do not involve experimental research on class imbalanced fraud prediction under decentralized conditions.

A widely discussed topic in FinTech literature is smart finance. Current developments in FinTech prove that AI can be used in payment automation, credit evaluation, risk analysis, and transaction intelligence [18]. Smart payment systems based on AI can be used for transaction flow monitoring, cash-flow optimization, and reducing imbalances in digital trade [19]. All these applications show that AI plays an increasingly important role in financial security and risk management. Nonetheless, many AI-based payment and risk systems suffer from several limitations, such as sensitive nature of financial data and its decentralization, class imbalance, and lack of explainability of black-box models. When compared to the existing research, the current methodology addresses several limitations. First, the study evaluates a federated-style simulation by partitioning the PaySim dataset into five simulated clients and training client-level models separately. This design illustrates decentralized data separation but does not constitute a complete privacy-preserving federated learning protocol. Second, severe class imbalance is addressed through class weighting, random undersampling, threshold optimization, and evaluation metrics appropriate for rare-event classification. Third, leakage-aware feature selection is applied by excluding balance-related attributes. Fourth, model-level interpretability is provided through global feature-importance analysis. Fifth, security mechanisms such as secure aggregation, differential privacy, client authentication, and poisoning detection are identified as future deployment enhancements.

Existing PaySim studies commonly focus on centralized classification accuracy or compare standard machine-learning models without systematically addressing leakage-prone balance variables, threshold optimization under extreme class imbalance, and decentralized client-level evaluation within the same experimental workflow. Published federated fraud detection studies generally rely on iterative parameter aggregation or neural-network-based FedAvg, whereas the present work evaluates a more limited prediction-level federated-style simulation using tree-based models. The contribution is therefore not a new federated optimization algorithm, but an integrated evaluation of leakage-aware feature screening, imbalance-sensitive metrics, threshold calibration, centralized benchmarking, and simulated decentralized aggregation.

3. Methodology

3.1 Research design

The proposed research employs a quantitative experiment-based approach that aims to create and validate a reliable and interpretable federated-style simulation architecture that can be utilized for fraud risk prediction in AI-powered FinTech platforms. The proposed research is computational and model-based, employing structured transaction-level financial data to emulate fraud prediction in both centralized and decentralized learning scenarios. This is done through an applied machine learning process where a publicly available financial fraud dataset is collected, pre-processed, split into simulated FinTech clients, and then employed to train fraud detection models. Four main steps have been identified for the proposed research process. First, there will be an understanding of the dataset along with the identification and mitigation of data leakage problems and feature selection. Second, there will be feature engineering and class-imbalance issues. Third, centralized machine learning models will be evaluated using imbalance-aware training and threshold tuning. Fourth, simulated decentralized learning is evaluated by splitting the dataset into multiple client partitions, training local models, comparing centralized, local, federated-style global, and ensemble aggregation results.

Figure 1 highlights the general research design and methodology used in this study. In the flowchart, it starts with a quantitative experimental research design that explores simulated decentralized learning setting for fraud risk prediction. The first step is about dataset understanding and data quality assessment using the PaySim dataset, which includes sample selection, quality checks, and leakage awareness in feature screening. Feature engineering and class-imbalance analysis involve creating temporal, monetary, and behavioural features, as well as analysing the

high fraud imbalance. Model development includes building a baseline model, learning with class imbalance, and threshold tuning. The next step focuses on federated learning simulation by dividing the dataset into five simulated FinTech clients and evaluating them using local models and federated-style aggregation. The methodology process concludes with evaluation and analysis of performance, explainability, and security of the algorithm by considering metrics such as precision, recall, F1 score, ROC-AUC, PR-AUC, and feature importance. In summary, Figure 1 illustrates the progress from data preparation towards trustworthy fraud risk prediction in decentralised FinTech platforms. Overall, the general objective of this study is not just about maximising predictive performance but also assessing fraud risk prediction in a realistic FinTech setting in which fraud incidents are scarce, data are distributed among various financial institutions, and decisions made by the model need to be interpretable. As a result, this study measures performance using metrics appropriate for imbalanced classification, such as precision, recall, F1 score, ROC-AUC, PR-AUC, false-positive rate, and confusion-matrix analysis.

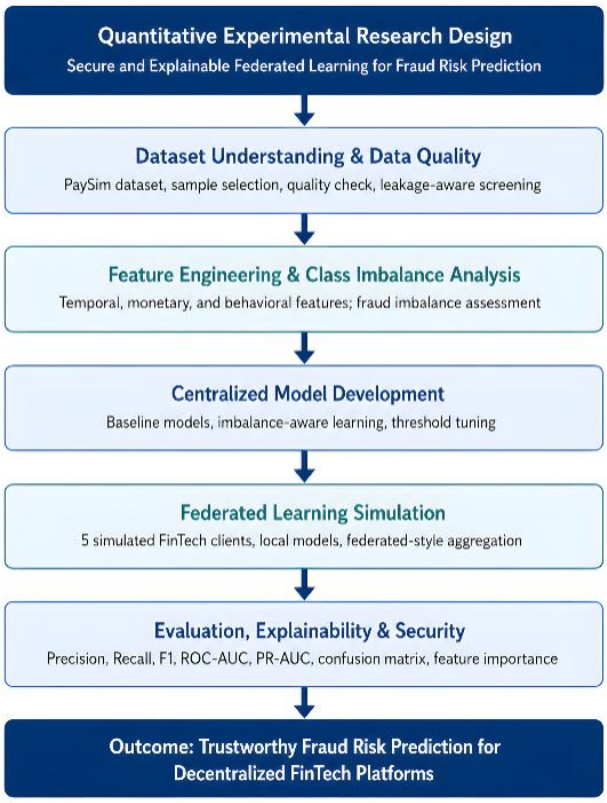

Figure 1. Methodological workflow for simulated decentralized learning setting-based fraud risk prediction

3.2 Data collection method

For this experiment, we have adopted the PaySim Synthetic Financial Dataset. It is an open-source dataset of mobile-money transaction data created using the PaySim simulator [20]. PaySim is a simulation of mobile money transaction data derived from aggregated patterns of actual transactions in a mobile money service. This dataset is appropriate for this study because it consists of transaction-level structured financial data with a binary fraud label. It has the following transaction attributes such as step, type, amount, nameOrig, oldbalanceOrg, newbalanceOrig, nameDest, oldbalanceDest, newbalanceDest, isFraud, and isFlaggedFraud. The step attribute is a measure of time where a step corresponds to one hour. Type attribute is the category of the transaction that includes CASH-IN, CASH-OUT, DEBIT, PAYMENT, and TRANSFER. Amount attribute is the monetary value of the transaction. Attributes of nameOrig and nameDest indicate the originator and the recipient respectively. The target attribute is isFraud, where 1 means fraud while 0 means legitimate transaction. While the entire PaySim dataset is huge, the sample used in this experimental process was 500,000 transactions. It had 11 original attributes, no missing values, and no duplicates. Data quality analysis indicated 233 cases of fraudulent transactions and 499,767 cases of legitimate transactions, thus generating a fraud rate of 0.0466%. This indicates the extreme imbalanced classes in fraud detection system.

3.3 Population and sampling

The target population of this research paper is comprised of transactions that take place in mobile-money and FinTech environments. In general terms, the population encompasses transactions made by customers, merchants, wallets, and participants of financial services. In terms of privacy and confidentiality considerations, direct access to the data of customers is not possible due to security and regulatory reasons. As a result, a synthetic and representative dataset that reflects the behavior of FinTech transactions is used. The PaySim dataset represents the sampling frame. For convenience and experimental purposes, a subset of 500,000 transactions was used for conducting experiments. The use of the sample helped to implement dataset analysis, model training, testing, imbalance management, federated learning partitioning, and other procedures. The highly imbalanced structure of fraud detection problems was also preserved.

For the purpose of model evaluation, the experiment was conducted on two subsets – training and testing. The stratified sampling procedure was implemented to ensure similar distributions of fraud and non-fraud cases in both subsets. In the model development sample, there were 240,000 training transactions and 60,000 testing transactions. In the test subset, there were 28 fraudulent transactions. For the federated-style simulation, only the 240,000-record training subset was partitioned into five simulated FinTech clients using deterministic originator-hash partitioning. Each client contained approximately 47,785 to 48,638 training records, with fraud rates ranging from 0.04176% to 0.05020%. The separate 60,000-record global test subset was excluded from all client-level training and was used only for final comparative evaluation.

3.4 Data preprocessing and leakage control

Preprocessing was conducted to clean the PaySim transaction data before using machine learning algorithms on it. The first preprocessing phase involved examining the data's columns' types, missing values, duplicate records, unique values, class distribution, and descriptive statistics. There were no missing values or duplicates in the 500,000 record sample. Leakage prevention measures were also undertaken prior to model building. According to the PaySim dataset documentation, fraudulent transactions are always canceled. As such, the balance-related features, oldbalanceOrg, newbalanceOrig, oldbalanceDest, and newbalanceDest, may provide predictive value based on the patterns of cancellations after the transaction takes place, leading to artificial data leakage. To mitigate this problem, the balance-related features were dropped from the predictive model. The feature isFlaggedFraud was also removed from

the improved model since there was no variance for this feature in the current sample.

The selected improved features include step, hour_of_day, day, type, amount, log_amount, is_high_value, dest_is_merchant, orig_txn_count, and dest_txn_count.

3.5 Feature engineering

Feature engineering was performed for better representation of transaction behavior. Features that captured temporal information were obtained from the step variable. The feature hour_of_day was calculated using step % 24, which captured transaction behavior on an hourly basis, whereas day was obtained using step // 24, which captured transaction behavior on a daily basis. Since the transaction amounts were generally skewed, a logarithm transformation was performed using log_amount = log1p(amount). The feature is_high_value was generated based on whether the transaction was greater than the 95th percentile of transaction amount. The feature dest_is_merchant was generated using the condition if the destination account number started with the letter "M," which indicated that the recipient was a merchant in the PaySim system. Frequency behavioral features were generated using orig_txn_count, which represented the number of times the originator made transactions, while dest_txn_count represented the number of times the recipient received transactions.

3.6 Data analysis techniques

The data analysis process followed a multi-step machine learning pipeline. First, an exploratory data analysis was done to check for class imbalance, transaction types distribution, distribution of fraud in different transaction types, transaction amounts distribution, and temporal distribution of fraud activity. It was found that frauds happened only in CASH_OUT and TRANSFER transactions, whereas no frauds occurred in CASH_IN, DEBIT, and PAYMENT transactions. Thus, the findings confirm the behavioral hypothesis according to which fraud in PaySim involved fund transfer and cash-out operations. Second, centralized benchmark models were trained using Logistic Regression, Random Forest, and class-weighted XGBoost. The XGBoost model was included as a stronger gradient-boosting baseline for structured and highly imbalanced transaction data. Precision, recall, F1-score, ROC-AUC, PR-AUC, and false-positive rate were prioritized instead.

Third, models were developed using the imbalance-aware approach. Class-weighted Logistic Regression, class-weighted Random Forest, random undersampling, and threshold tuning were applied. Threshold tuning was especially necessary because the threshold of 0.50 was not the best option for the fraud detection problem. In particular, threshold tuning helped find the optimal decision threshold providing the best balance between precision and recall. Models' performances were analyzed using F1-score, ROC-AUC, PR-AUC, false-positive rate, and confusion matrices. Finally, a federated-style simulation was carried out. Models were trained separately for each local client and compared to centralized and federated-style models. Since the FedAvg technique cannot be directly applied to train Random Forest models, a federated-style ensemble averaging was done to combine client-level prediction probabilities.

3.7 Experimental Configuration and Reproducibility

The experiments were implemented in Python using NumPy, pandas, matplotlib, scikit-learn, and XGBoost. A 500,000-record sample containing 11 original variables was loaded from the PaySim file PS_20174392719_1491204439457_log.csv. After feature engineering, 300,000 records were sampled using random_state=42 for model development. An 80:20 stratified split was applied using train_test_split(test_size=0.2, random_state=42, stratify=y), producing 240,000 training records and 60,000 test records, including 28 fraud cases. The target variable was isFraud. The predictive features were step, hour_of_day, day, type, amount, log_amount, is_high_value, dest_is_merchant, orig_txn_count, and dest_txn_count. The balance variables oldbalanceOrg, newbalanceOrig, oldbalanceDest, and newbalanceDest were excluded to reduce leakage risk, while isFlaggedFraud was removed because it showed no variation in the selected sample. Transaction type was encoded using OneHotEncoder(handle_unknown="ignore"), numerical variables were standardized using StandardScaler, and all preprocessing was implemented through ColumnTransformer and model pipelines.

The Logistic Regression model used max_iter=1000, class_weight="balanced", and n_jobs=-1. The Improved Random Forest used 200 trees, unrestricted depth, min_samples_leaf=2, class_weight="balanced_subsample", random_state=42, and n_jobs=-1. For the undersampling experiment, the majority class was randomly reduced to a 20:1 majority-to-minority ratio using random_state=42, resulting in 111 fraud and 2,220 non-fraud training records. The undersampled Random Forest used the same tree settings but without class weighting. A class-weighted XGBoost model was also evaluated using 300 estimators, max_depth=4, learning_rate=0.05, subsample=0.8, colsample_bytree=0.8, a binary logistic objective, a positive-class weight equal to the non-fraud-to-fraud ratio in the training set, random_state=42, and n_jobs=-1. Decision thresholds from 0.01 to 0.99 were evaluated in increments of 0.01. The F1-optimal thresholds were 0.33 for the Improved Random Forest and 0.95 for XGBoost. Performance was assessed using accuracy, precision, recall, F1-score, ROC-AUC, PR-AUC, false-positive rate, confusion-matrix values, and the resulting counts of true positives, false positives, false negatives, and true negatives.

For the federated-style simulation, only the 240,000-record training subset was divided into five simulated FinTech clients through deterministic originator-hash partitioning. Client IDs were generated using pd.util.hash_pandas_object(nameOrig, index=False) % 5, ensuring that all transactions from the same originator were assigned to one client. The separate 60,000-record global test subset was excluded from client-level training. Each client trained an independent Random Forest using its locally assigned training records, and all client models were subsequently evaluated on the same untouched global test subset. For Federated Ensemble Aggregation, the five independently trained client models generated fraud probabilities for the common global test subset. The probabilities were combined through simple unweighted averaging, after which the selected threshold was applied. No client training records, model parameters, gradients, or tree structures were exchanged during local model fitting. The method therefore represents prediction-level ensemble aggregation rather than FedAvg or parameter-level federated optimization.

3.8 Explainability, security, and evaluation

Explainability in this study was implemented through global feature-importance analysis from the Improved Random Forest model. This method identifies which variables

contributed most strongly to the overall fraud-risk prediction process, including transaction amount, log-transformed transaction amount, time-related variables, transaction type, and recipient-account activity. Therefore, the explainability component should be interpreted as model-level interpretability rather than transaction-level local explanation. Local explanation methods such as SHAP and LIME were not implemented in the present experiment and are identified as future extensions. From a security and privacy perspective, the study follows a simulated decentralized design in which raw transaction records remain within client-level partitions during local model training. This design reduces dependence on centralized raw-data aggregation. However, cryptographic secure aggregation, differential privacy, client authentication, poisoning detection, encryption protocols, and audit logging were not implemented or experimentally evaluated. These mechanisms are treated as future deployment enhancements rather than tested contributions of the present study. In summary, the methodological framework should be understood as a leakage-aware and imbalance-aware federated-style simulation with global feature-importance-based explainability. It does not represent a completely secure federated learning implementation.

4. Results

4.1 Dataset presentation and dataset characteristics

The experiment was carried out on a sample of 500,000 records drawn from the PaySim Synthetic Financial Dataset. The dataset comprised eleven features that represented transaction time, transaction type, transaction amount, originator account, destination account, balance features, and fraud indicators. Preliminary data exploration revealed that the dataset did not have any missing values and duplicates, suggesting that the dataset was ready for preprocessing and modeling. However, the class distribution for the target variable indicated an extremely imbalanced dataset. Of the total 500,000 records, only 233 were classified as fraud transactions, whereas 499,767 were classified as non-fraud transactions. This resulted in a fraud percentage of around 0.0466%. As depicted in Table 1, the PaySim sample dataset has 500,000 rows without any missing or repeated values. Nevertheless, only 233 instances are fraudulent, representing 0.0466% of all records. It can be observed from the above findings that there is a class imbalance in the dataset and fraud metrics should be used.

Table 1. Data quality and class distribution summary

Data Quality Indicator	Result
Total records	500,000
Total original variables	11
Missing values	0
Duplicate records	0
Fraud records	233
Non-fraud records	499,767
Fraud rate	0.0466%
Imbalance ratio	1 fraud : 2144.92 non-fraud

The class distribution in the PaySim data is displayed in Figure 2 using a logarithmic scale. It is evident from the figure that there is a tremendous imbalance between the non-fraud and fraud classes. In total, there were 500,000 cases, out of which 499,767 were classified as non-fraud and 233 were classified as fraud cases. It is through the use of logarithmic scaling that the imbalance in class distribution is revealed, making it easier to deal with the problem of rare event prediction of fraud.

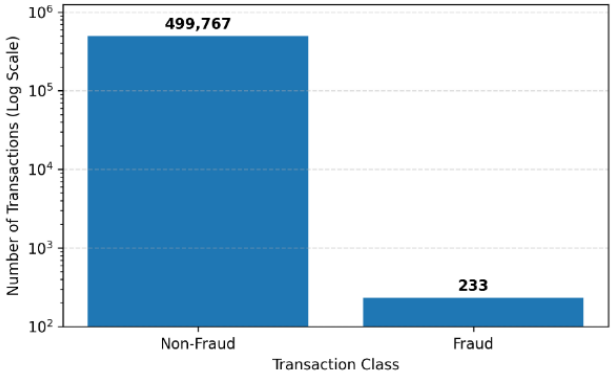

Figure 2. Fraud and Non-Fraud Class Distribution on Logarithmic Scale

4.2 Transaction type patterns and fraud distribution

In the exploratory data analysis, it was established that there was an unequal distribution of transaction types. In terms of frequency, the top five transaction types were CASH_OUT, PAYMENT, CASH_IN, TRANSFER, and DEBIT, in that order. There were frauds observed in only two types of transactions, namely CASH_OUT and TRANSFER. Of these, CASH_OUT had 121 fraud transactions, while TRANSFER had 112 fraud transactions. Table 2 shows that fraud occurs only in CASH_OUT and TRANSFER transactions. CASH_OUT contains the highest number of fraud cases, while TRANSFER has the highest fraud rate. This suggests that fraud in the dataset is mainly linked to fund transfer and cash-out behavior. Figure 3 illustrates the total number of fraudulent transactions by each transaction type in the examined PaySim sample. In total, there were 121 and 112 fraudulent cases for CASH_OUT and TRANSFER transactions, respectively, while CASH_IN, DEBIT, and PAYMENT had zero cases of fraud. It can be seen that fraud in the given dataset is highly correlated with money transfers and withdrawals, which implies that the behavior of fraudulent agents is aimed at moving and withdrawing money from compromised accounts.

Table 2. Fraud distribution by transaction type

Transaction Type	Total Transactions	Fraud Transactions	Fraud Rate
CASH_OUT	182,316	121	0.0664%
TRANSFER	40,730	112	0.2750%
CASH_IN	109,319	0	0.0000%
DEBIT	3,603	0	0.0000%
PAYMENT	164,032	0	0.0000%

4.3 Feature engineering and leakage-aware screening

Prior to model training, balance-related leakage variables were left out of the predictive model. The balance variables that were not used in the modeling process included oldbalanceOrg, newbalanceOrig, oldbalanceDest, and newbalanceDest. Balance variables were not used since the documentation provided by the PaySim dataset explains that fraudulent transactions are reversed, and thus, the balance variables could exhibit leakage from the transaction itself.

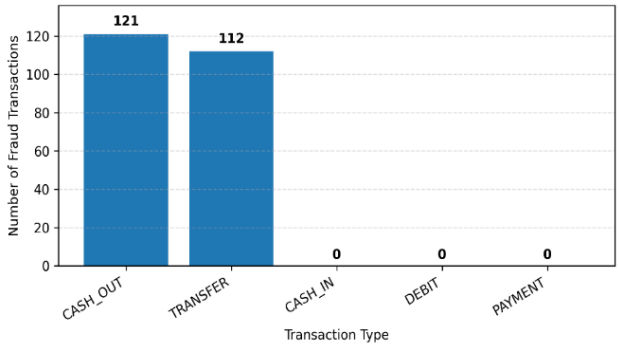

Figure 3. Fraud transactions across mobile-money transaction types

In addition, the isFlaggedFraud variable was left out since there was no variation found among 500,000 observations. The final list of engineered features included temporal features, monetary features, and behavioral features. The temporal features included step, hour_of_day, and day. The monetary features included amount, log_amount, and is_high_value. The behavioral/account activity features included dest_is_merchant, orig_txn_count, and dest_txn_count. The above features are consistent with the aim of the study. Table 3 presents the final features used for model training after leakage-aware screening. The selected features capture temporal, monetary, and behavioral transaction patterns while excluding balance-related variables that could introduce data leakage. Figure 4 shows that monetary features, especially amount and log_amount, were the strongest predictors of fraud risk. Time-related features and destination activity also contributed meaningfully, indicating that fraud prediction depends not only on transaction value but also on temporal and behavioral transaction patterns.

Table 3. Final engineered features used for model training

Feature Group	Features	Purpose
Temporal features	step, hour_of_day, day	Capture time-based transaction behavior
Monetary features	amount, log_amount, is_high_value	Capture transaction value and high-value risk
Behavioral features	dest_is_merchant, orig_txn_count, dest_txn_count	Capture recipient type and transaction frequency
Target variable	isFraud	Binary fraud-risk label

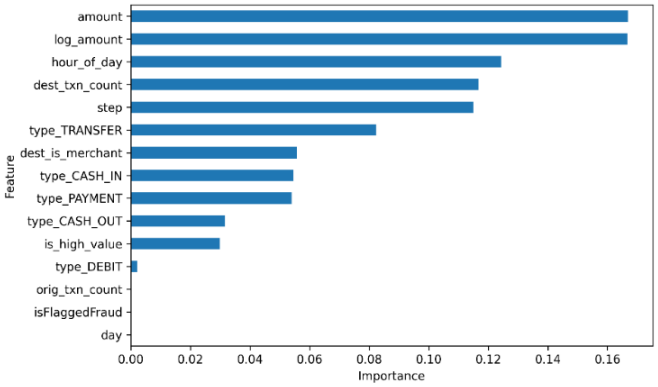

Figure 4. Feature importance analysis of the improved random forest fraud prediction model

4.4 Centralized model results and imbalance-aware optimization

The first batch of centralized algorithms demonstrated clear signs of the precision-recall trade-off. Specifically, Logistic Regression had very high recall but almost no precision. On the other hand, Random Forest had good precision and accuracy but poor recall. Since there were very few fraud instances in the dataset, the third stage of model development applied class weighting, random undersampling, and threshold tuning. The Improved Random Forest at a threshold of 0.33 achieved the highest F1-score of 0.3019, with a precision of 0.3200, recall of 0.2857, ROC-AUC of 0.9211, and PR-AUC of 0.1700. It correctly identified 8 of the 28 fraud cases while producing 17 false-positive alerts. The class-weighted XGBoost model provided a stronger gradient-boosting baseline. At its F1-optimized threshold of 0.95, XGBoost achieved a precision of 0.2250, recall of 0.3214, F1-score of 0.2647, ROC-AUC of 0.9458, and PR-AUC of 0.2034. XGBoost identified 9 fraud cases but generated 31 false positives. Therefore, XGBoost demonstrated better fraud-ranking capability and slightly higher recall, whereas the Improved Random Forest provided the strongest balance between precision and recall and remained the selected centralized model. Table 4 compares the centralized imbalance-aware models. The Improved Random Forest at a threshold of 0.33 achieved the highest F1-score and provided the most balanced precision-recall trade-off. XGBoost achieved higher ROC-AUC, PR-AUC, and recall, demonstrating stronger ranking capability, but it produced more false positives and a lower F1-score.

Table 4. Centralized and imbalance-aware model performance

Model	Threshold	Accuracy	Precision	Recall	F1-score	ROC-AUC	PR-AUC	TP	FP	FN
Improved Random Forest	0.33	0.9994	0.3200	0.2857	0.3019	0.9211	0.1700	8	17	20
Random Forest with Undersampling	0.50	0.9947	0.0550	0.6429	0.1014	0.9450	0.0888	18	309	10
Improved Logistic Regression	0.99	0.9978	0.0579	0.2500	0.0940	0.9482	0.1126	7	114	21
Improved Logistic Regression	0.50	0.8296	0.0025	0.9286	0.0051	0.9482	0.1126	26	10,223	2
XGBoost with Class-Imbalance Weighting	0.95	0.9992	0.2250	0.3214	0.2647	0.9458	0.2034	9	31	19

The undersampled Random Forest increased recall substantially at the default threshold but also generated a much larger number of false-positive alerts. Figure 5 is the performance plot of the Improved Random Forest algorithm at threshold 0.33. The performance results are 59,955 true negatives, 8 true positives, 17 false positives, and 20 false negatives. The confusion matrix indicates high non-fraud specificity and a low false-positive count. However, fraud recall remained limited because the model detected only 8 of the 28 fraud cases.

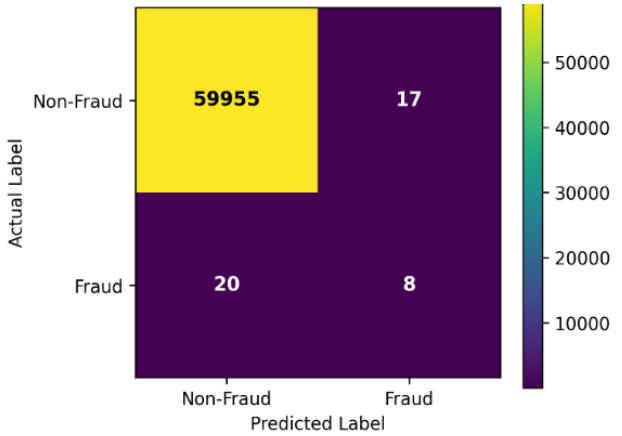

Figure 5. Confusion matrix of the improved random forest model at threshold 0.33

4.5 Federated-style simulation results

To evaluate simulated decentralized fraud-risk prediction, the 240,000-record training subset was divided into five simulated FinTech clients using deterministic originator-hash partitioning. All transactions associated with the same originator account were assigned to the same client. The separate 60,000-record global test subset was excluded from all client-level training and was used only for final comparative evaluation. Each client contained approximately 47,785 to 48,638 training records, with fraud rates ranging from 0.04176% to 0.05020%. Table 5 shows that the training records were distributed relatively evenly across the five simulated FinTech clients. The client fraud rates were also similar, indicating that the simulation was close to an independent and identically distributed, or near-IID, setting. Nevertheless, fraud remained extremely rare within every client partition, which continued to create a severe class-imbalance problem for local model training. Each client trained an independent Random Forest model using only its locally assigned training records. For Federated Ensemble Aggregation, the five trained client models generated fraud-risk probabilities for the same untouched global test subset. These probabilities were combined using simple unweighted averaging, after which the selected decision threshold was applied to obtain the final classifications. No client training records, model parameters, gradients, or tree structures were exchanged during local model fitting.

For experimental comparison, the trained models were evaluated centrally on the same shared held-out test records. Therefore, Federated Ensemble Aggregation represents prediction-level ensemble aggregation in a simulated decentralized setting rather than FedAvg-based or parameter-level federated optimization. Figure 6 compares the PR-AUC values of the centralized Improved Random Forest, the Average Local Client Model, and Federated Ensemble Aggregation on the same untouched global test set. Federated Ensemble Aggregation achieved the highest PR-AUC of 0.2699, followed by the centralized Improved Random Forest at 0.1700 and the Average Local Client Model at 0.1205. Although the ensemble demonstrated stronger fraud-ranking performance, its recall remained low at 0.1429. Therefore, PR-AUC should be interpreted together with threshold-level precision, recall, and F1-score when evaluating rare-event fraud detection.

The centralized Improved Random Forest achieved the highest F1-score of 0.3019, with precision of 0.3200 and recall of 0.2857. Federated Ensemble Aggregation achieved precision of 1.0000, recall of 0.1429, F1-score of 0.2500, ROC-AUC of 0.9543, and PR-AUC of 0.2699. It identified 4 of the 28 fraud cases and produced no false-positive alerts. The Average Local Client Model achieved an F1-score of 0.1569 and PR-AUC of 0.1205. These results show that the ensemble was highly conservative, while the centralized model provided the strongest overall precision–recall balance. Table 6 compares all models on the same untouched global test set. The centralized Improved Random Forest achieved the highest F1-score and provided the strongest balance between precision and recall.

Table 5. Federated client distribution

Client	Total Records	Fraud Records	Fraud Rate
0	48,638	23	0.04729%
1	47,808	24	0.05020%
2	47,888	20	0.04176%
3	47,785	22	0.04604%
4	47,881	22	0.04595%

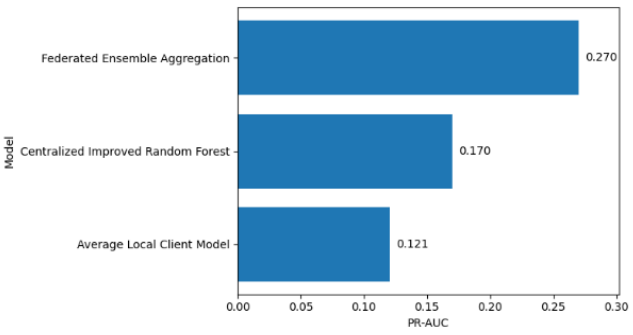

Figure 6. Centralized, local, and federated-style model comparison based on PR-AUC

Table 6. Centralized and simulated decentralized model comparison on the common global test set

Model	Precision	Recall	F1-score	ROC-AUC	PR-AUC
Centralized Improved Random Forest	0.3200	0.2857	0.3019	0.9211	0.1700
Average Local Client Model	0.2593	0.1143	0.1569	0.8877	0.1205
Federated Ensemble Aggregation	1.0000	0.1429	0.2500	0.9543	0.2699

Federated Ensemble Aggregation achieved the highest precision, ROC-AUC, and PR-AUC, but its low recall indicates conservative fraud classification. The Average Local Client Model produced the lowest F1-score and PR-AUC, reflecting the difficulty of training stable models when each client contains very few fraud records. Figure 7 compares the F1-scores of the centralized Improved Random Forest, the Average Local Client Model, and Federated Ensemble Aggregation. The centralized model achieved the highest F1-score of 0.3019, followed by Federated Ensemble Aggregation at 0.2500 and the Average Local Client Model at 0.1569. The ensemble's lower recall indicates that further threshold calibration would be required before operational deployment. However, it can be seen from Figure 7 that the recall rate of federated ensemble aggregation is lower than other models, indicating that threshold calibration will be required during deployment of fraud detection models in FinTech organizations.

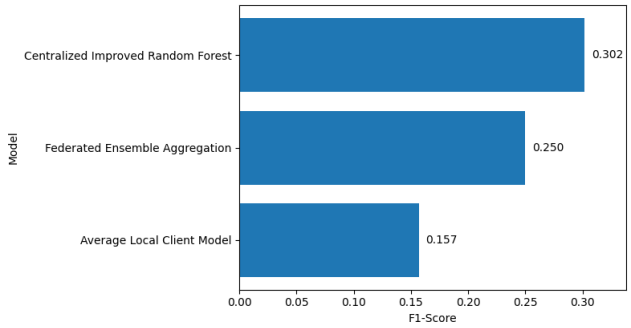

Figure 7. Centralized, local, and federated-style model comparison based on F1-score

4.6 Key findings, patterns, and trends

Four key findings can be extracted from the results. Firstly, fraud detection in the PaySim dataset is extremely unbalanced, with fraud comprising less than 0.05% of all transactions. Secondly, fraud behavior is limited to CASH_OUT and TRANSFER transactions, suggesting that fraudulent financial flows and withdrawals play a crucial role in the fraud risk. Thirdly, adjusting the threshold is vital since the default value does not provide optimal performance in terms of the balance between precision and recall. Finally, the prediction-level ensemble provided stronger ranking performance than the average local-client model, although the centralized Improved Random Forest retained the highest F1-score. The most significant pattern that can be observed here is the tradeoff between precision and recall. Models with high recall find more fraudulent transactions but also generate many false positives, while high-precision models decrease false positives but fail to identify some of the fraudulent activities. The selected Improved Random Forest at threshold 0.33 yields the most balanced outcome, while the federated ensemble model shows the highest values for precision and PR-AUC but relatively low recall.

5. Discussion

Firstly, it was discovered that class imbalance, quality of features, and choice of threshold have a significant effect on fraud risk prediction in AI-based FinTech platforms. In the PaySim sample, there were only 233 frauds out of 500,000 transactions, representing a fraud rate of 0.0466%. This proves that fraud detection is indeed a rare event classification task. Fraud cases occurred primarily in the CASH_OUT and TRANSFER transaction types, which suggests that any suspicious movements or withdrawals can indicate fraud risk [7]. For the centralized model, a trade-off between precision and recall can be observed. While the Logistic Regression model had high recall, it produced many false positive predictions. On the other hand, the improved Random Forest model, after threshold tuning, had the optimal results, as it had the highest F1-score. Therefore, threshold optimization is necessary when class imbalance is severe. This finding confirms the importance of anomaly detection in high-frequency financial transactions [1].

The addition of XGBoost provided a stronger benchmark for evaluating the proposed leakage-aware feature set. XGBoost achieved higher ROC-AUC and PR-AUC than the Improved Random Forest and detected one additional fraud case. However, this improvement was accompanied by a reduction in precision and an increase in false-positive alerts. The result indicates that gradient boosting may rank rare fraud cases more effectively, while the threshold-optimized Random Forest provides a more balanced operational trade-off when both missed fraud cases and unnecessary alerts are considered. In the federated-style simulation, the independently trained local-client models showed limited stability because each client contained very few fraud cases. Federated Ensemble Aggregation achieved the highest PR-AUC of 0.2699 and a precision of 1.0000 but detected only 4 of the 28 fraud cases, resulting in recall of 0.1429. These findings indicate that prediction-level averaging improved fraud-ranking performance and reduced false-positive alerts, but further threshold calibration would be necessary to improve fraud detection sensitivity. The results should be interpreted within the present simulated and near-IID setting rather than as evidence of a complete federated learning implementation.

Feature-importance analysis showed that transaction amount, log-transformed amount, time-related variables, transaction type, and destination activity were among the most influential predictors. This provides model-level interpretability by identifying the variables that contributed most strongly to the Random Forest fraud-risk predictions. However, this form of explainability is global and does not explain individual transaction-level decisions. For practical regulatory and analyst-facing use, future work should add local explanation methods such as SHAP or LIME.

The framework has limited privacy-related implications because the simulated client partitions are trained separately and raw transaction records are not pooled into a single centralized training dataset. However, this should not be interpreted as a full security implementation. Secure aggregation, differential privacy, encryption, client authentication, poisoning detection, and audit logging were not implemented or evaluated in the current experiment. These mechanisms are better understood as future deployment requirements for a production-level secure federated fraud-detection system.

This study has several limitations that should be considered when interpreting the findings. First, the experiment used a 500,000-record synthetic subsample from the PaySim dataset rather than the full PaySim dataset or real institutional transaction data. Although PaySim is useful for controlled fraud-detection experiments, real FinTech environments may contain more complex customer behavior, institution-specific transaction patterns, and changing fraud strategies. Second, the federated component was implemented as a federated-style simulation rather than a true FedAvg-based or cryptographically secured federated learning protocol. The study used client partitioning, local

model training, and prediction-level aggregation, but it did not implement parameter exchange, iterative communication rounds, secure aggregation, or differential privacy. Third, security mechanisms such as encryption, client authentication, differential privacy, secure aggregation, poisoning detection, and audit logging were discussed as future deployment requirements and were not experimentally implemented or evaluated. Fourth, explainability was limited to global Random Forest feature-importance analysis. This identifies influential variables at the model level but does not provide transaction-level explanations for individual fraud decisions. Future work should incorporate local explanation methods such as SHAP or LIME. Fifth, the five simulated client partitions had similar record counts and fraud rates, making the experimental setting closer to an IID simulation than a heterogeneous cross-institution FinTech environment. In real deployments, clients may differ substantially in customer base, transaction-type distribution, fraud prevalence, and risk behavior. Therefore, future research should evaluate non-IID client partitions, real or larger transaction datasets, stronger baselines such as gradient-boosted models, and fully implemented privacy-preserving federated learning protocols.

The findings provide several practical implications for FinTech fraud-monitoring systems. First, CASH_OUT and TRANSFER transactions should receive prioritized monitoring because all fraud cases in the examined PaySim sample occurred within these transaction categories. Second, decision thresholds should be calibrated separately for each institution because fraud prevalence, customer behavior, analyst capacity, and the operational cost of false alerts may differ across banks, wallets, and payment-service providers. Third, increasing recall can reduce missed fraud but may also create substantial alert fatigue, manual-review costs, and customer inconvenience. Conversely, highly precise models may overlook a large proportion of fraudulent transactions. Finally, the higher PR-AUC achieved by Federated Ensemble Aggregation should not alone be treated as sufficient evidence for deployment, because its recall of 0.1429 means that it detected only 4 of the 28 fraud cases at the selected threshold. Institution-specific threshold calibration and cost-sensitive evaluation are therefore necessary before operational adoption.

6. Conclusion

This study evaluated a leakage-aware and imbalance-aware federated-style simulation for fraud-risk prediction in AI-driven FinTech platforms. Using a 500,000-record PaySim sample, the study demonstrated that mobile-money fraud prediction is a highly imbalanced rare-event classification problem, with fraudulent transactions concentrated in CASH_OUT and TRANSFER operations. Based on this study's findings, leakage detection, feature selection, class imbalance handling, and threshold tuning are essential factors to consider when performing mobile-money fraud predictions. The Improved Random Forest achieved the highest centralized F1-score of 0.3019. Federated Ensemble Aggregation achieved the highest PR-AUC of 0.2699 and produced no false-positive alerts, but its recall of 0.1429 indicates that it missed most fraud cases. Therefore, the centralized model provided the strongest overall precision-recall balance, while the ensemble demonstrated conservative but stronger ranking behavior. Besides, feature importance analysis showed that the transaction amount, time-related factors, transaction type, and destination account activities are key factors influencing the outcome of fraud predictions. The combination of model accuracy and interpretability is highly valuable for this application domain because model decisions need to be transparent, traceable, and operationally feasible. Additionally, the decentralized model training reduces reliance on centralized raw data processing while retaining decent fraud-risk prediction performance. As such, the results suggest that simulated decentralized learning can reduce dependence on centralized raw-data processing while maintaining comparable fraud-risk prediction performance. However, the present study does not implement FedAvg, secure aggregation, differential privacy, or transaction-level SHAP/LIME explanations; these remain important directions for future work. From an applied-technology perspective, this research provides an experimental foundation for decentralized and interpretable fraud-risk modelling without claiming a complete security or production-level federated learning implementation. This enhances the practical significance of this study for regulatory bodies, financial institutions, and technology providers looking to develop trustworthy AI-driven fraud monitoring systems. Possible areas of future research include the implementation of FedAvg-based neural federated learning, cryptographic measures for privacy protection, testing on more extensive real-world transaction datasets, and SHAP/LIME-based transaction-level explanations.

Ethical issue

The authors are aware of and comply with best practices in publication ethics, specifically with regard to authorship (avoidance of guest authorship), dual submission, manipulation of figures, competing interests, and compliance with policies on research ethics. The authors adhere to publication requirements that the submitted work is original and has not been published elsewhere. All survey participants provided informed consent prior to participation, and the anonymity and confidentiality of all respondents were strictly maintained throughout the research process.

Data availability statement

The manuscript contains all the data. However, additional data will be provided by the corresponding author upon reasonable request.

Conflict of interest

The authors declare no potential conflict of interest.

References

[1] H. A. Al Dulaimi et al., “AI-driven behavioral anomaly and fraud detection models for real-time high-frequency financial transactions in FinTech systems,” in Proc. 2025 3rd Int. Conf. Cyber Resilience (ICCR), Dubai, United Arab Emirates, 2025, pp. 1–7, doi: 10.1109/ICCR67387.2025.11292440.

[2] M. K. Sandaruwan, D. V. D. S. Abeysinghe, and L. D. S. B. Weerasinghe, “AI-enhanced microservice architecture in the financial domain: A systematic review of security and performance,” in Proc. 2026 IEEE Int. Research Conf. Smart Computing and Systems Engineering (SCSE), vol. 9, Mar. 2026, pp. 1–6.

[3] K. Jayabalan, S. Parmsivan, G. Sunkara, M. Parikh, P. Challa, and V. Nutalapati, “Innovative framework for secure and scalable web and mobile application development in fintech: A user-centric and AI-driven approach,” in Proc. 2025 5th Int. Conf. Ubiquitous

Computing and Intelligent Information Systems (ICUIS), Nov. 2025, pp. 1499–1505.
[4] A. Tripathi, "Transforming finance with AI: A capability-based framework for intelligent banking," in Proc. 2025 Int. Conf. Responsible, Generative and Explainable AI (ResGenXAI), Sep. 2025, pp. 1–5.
[5] V. Sampathkumar and K. Veeras, "Adversarial attacks on FinTech AI models: Threats and mitigation techniques," in Proc. 2025 IEEE Int. Carnahan Conf. Security Technology (ICCST), Oct. 2025, pp. 1–7.
[6] S. Chandra, S. Muniraj, S. Y. Reddy, V. Raju, B. R. Kumar, and S. K. Katta, "Blockchain and AI integration in fintech: Securing financial transactions and ensuring transparency," in Proc. 2025 IEEE 4th World Conf. Applied Intelligence and Computing (AIC), Jul. 2025, pp. 257–261.
[7] J. Fan, L. K. Shar, R. Zhang, Z. Liu, W. Yang, D. Niyato, and K. Y. Lam, "Deep learning approaches for anti-money laundering on mobile transactions: Review, framework, and directions," IEEE Internet of Things Journal, 2026.
[8] Z. M. Guria, I. Rahman, S. Kowser, N. Morshed, R. S. Shammah, and O. Faruq, "AI-driven risk management and fraud prevention in digital finance as a catalyst for US economic resilience and global financial leadership," in Proc. 2025 IEEE 19th Int. Conf. Open Source Systems and Technologies (ICOSST), Dec. 2025, pp. 1–8.
[9] P. Chinnasamy, A. R. Shaik, A. Akhila, Y. Tejasri, D. Vaishnavi, and D. P. Degala, "FinGuard—AI-powered credit risk assessment for the underbanked using machine learning," in Proc. 2025 Int. Conf. Future Technologies (ICFT), Nov. 2025, pp. 1–6.
[10] Y. Zuo, "Exploring the synergy: AI enhancing blockchain, blockchain empowering AI, and their convergence across IoT applications and beyond," IEEE Internet of Things Journal, vol. 12, no. 6, pp. 6171–6195, 2024.
[11] B. Madupati, R. K. Cherukuri, P. K. Keer, N. Naik, N. N. Prabhu, and S. J. Kidwai, "A secure cloud–AI collaborative framework for financially intelligent and autonomous manufacturing enterprises," in Proc. 2026 Innovations in Machine, Engineering, and Digital Conference (IMED), Mar. 2026, pp. 1–7.
[12] M. F. Ansari, A. Mohammed, K. R. Janamolla, S. W. Khadri, S. A. Khader, and M. A. Raheem, "The double-edged sword: Navigating AI's role in banking cybersecurity," in Proc. 2025 Int. Conf. Transformative Computing Technologies (ICTCT), Sep. 2025, pp. 294–300.
[13] R. Lotus, L. R. Peram, S. K. R. Vangala, and S. V. Kusampudi, "A resilient cybersecurity framework for enhancing mobile application security with artificial intelligence," in Proc. 2025 4th Int. Conf. Automation, Computing and Renewable Systems (ICACRS), Dec. 2025, pp. 1879–1883.
[14] M. Rizinski, H. Peshov, K. Mishev, L. T. Chitkushev, I. Vodenska, and D. Trajanov, "Ethically responsible machine learning in fintech," IEEE Access, vol. 10, pp. 97531–97554, 2022.
[15] V. R. Pasam, B. Krishnan, R. R. Charla, R. Somayajula, and S. M. Veerapaneni, "AI-enhanced data engineering for complex workflows: Lessons from retail intelligence and financial risk models," in Proc. 2025 Int. Conf. Computing Technologies & Data Communication (ICCTDC), Jul. 2025, pp. 1–8.
[16] V. Jaiswal, A. Shahi, and A. Kaushal, "Integration of blockchain with artificial intelligence for financial security: A comprehensive review," in Proc. 2025 IEEE 7th Int. Conf. Computing, Communication and Automation (ICCCA), Nov. 2025, pp. 1–6.
[17] R. G. Franklin, S. Kaur, B. Kumari, Z. Haixia, M. Gupta, and T. Agarwal, "Cyber-resilient financial management framework using blockchain, AI, and cloud-oriented auditing," in Proc. 2025 IEEE 1st Int. Conf. Smart Innovations in Systems, Infrastructure, Mechanical, Power, AI and Computing Technologies (SISIMPACT), Nov. 2025, pp. 815–820.
[18] P. D. Akre, U. Pacharaney, and W. Siraskar, "Smart finance: An overview of artificial intelligence integration in fintech," in Proc. 2024 2nd DMIHER Int. Conf. Artificial Intelligence in Healthcare, Education and Industry (IDICAIEI), Nov. 2024, pp. 1–6.
[19] K. Saradhi and S. Kaliappan, "AI-based smart payment systems for preventing cash flow imbalances in trade," in Proc. 2025 IEEE 1st Int.
[20] E. A. Lopez-Rojas, A. Elmir, and S. Axelsson, "PaySim: A financial mobile money simulator for fraud detection," in Proc. 28th European Modeling and Simulation Symposium (EMSS), Larnaca, Cyprus, 2016.