

PRIVACY-PRESERVING CRYPTOCURRENCY FRAUD DETECTION USING FEDERATED LEARNING

Sweta Kahurke¹, Harsha Jain², Shifra Sheikh³, Srushti Pillare⁴, Vishakha
Kandrikar⁵

¹Professor, ²³⁴⁵Student

¹²³⁴⁵Department of Artificial Intelligence,

¹²³⁴⁵JD College of Engineering and Management, Nagpur, India

Abstract: Decentralized finance (DeFi) platforms have experienced a rapid increase in fraudulent activities such as price manipulation, wash trading, and anomalous transaction behavior, while traditional centralized fraud detection methods remain unsuitable due to privacy and regulatory constraints. This study proposes a privacy-preserving fraud detection framework using federated learning, enabling multiple decentralized entities to collaboratively train a machine learning model without sharing raw transaction data. A real-world decentralized exchange (DEX) dataset containing over 100,000 transactions is preprocessed and enhanced through feature engineering techniques capturing swap rate deviations, transaction volume anomalies, and temporal patterns. In the absence of labeled fraud data, a heuristic-based labeling approach is employed to simulate realistic fraud scenarios. A Logistic Regression model is trained across multiple distributed client nodes, with model parameters aggregated using the Federated Averaging (FedAvg) algorithm over several communication rounds. The experimental findings show that the federated model delivers results similar to centralized methods while preserving data privacy, proving it to be an efficient solution for secure and scalable fraud detection in decentralized financial environments.

Keywords: Federated Learning, Fraud Detection, Decentralized Finance (DeFi), Cryptocurrency Transactions, Privacy-Preserving Machine Learning, Distributed Learning, Logistic Regression, Anomaly Detection, Blockchain Analytics, Federated Averaging (FedAvg)

1. INTRODUCTION

The expansion of decentralized finance (DeFi) and blockchain-driven trading platforms has reshaped how digital transactions are conducted. Decentralized exchanges (DEXs) like Uniswap and SushiSwap facilitate direct peer-to-peer trading without relying on a central authority, promoting openness and ease of access. At the same time, the lack of centralized oversight and the pseudonymous characteristics of blockchain activity have heightened the potential for fraud, such as market manipulation, wash trading, and unusual transaction patterns. These challenges pose serious threats to the reliability, security, and user trust in decentralized financial ecosystems.

Traditional fraud detection systems rely on centralized data aggregation and analysis to build accurate machine learning models. While effective in conventional financial systems, such approaches are not suitable for decentralized environments due to strict privacy constraints, regulatory limitations, and the reluctance of organizations to share sensitive transaction data. As a result, existing fraud detection models often suffer from limited data availability, reducing their ability to generalize across diverse transaction patterns.

To overcome these challenges, federated learning has gained attention as an effective approach that allows multiple parties to train models collaboratively without exchanging

raw data. In this method, each participant builds a local model using its own private dataset and shares only the learned parameters with a central server. The server then combines these updates to create a unified global model, preserving data privacy while benefiting from insights across distributed datasets.

In this paper, a federated learning-based framework is proposed for detecting fraudulent cryptocurrency transactions in decentralized environments. The system utilizes a real-world DEX transaction dataset and applies preprocessing and feature engineering techniques to extract meaningful behavioral patterns. A Logistic Regression model is trained in a federated setting, where model updates are aggregated using the Federated Averaging (FedAvg) algorithm. The proposed approach aims to achieve high detection performance while preserving data confidentiality.

The remainder of this paper is organized as follows:

Section 2 presents the literature review, Section 3 describes the proposed methodology, Section 4 discusses the implementation and experimental results, and Section 5 concludes the paper with future research directions.

2. LITERATURE REVIEW

2.1 Existing Approaches

Cryptocurrency fraud detection has been widely studied using various computational techniques, primarily focusing on identifying abnormal transaction behavior and suspicious financial activities. Machine learning-based approaches such as Logistic Regression, Random Forest, and Support Vector Machines have been extensively used for classification of transactions into fraudulent and legitimate categories. These models rely on historical transaction data and extracted features to detect patterns associated with fraud.

In addition to traditional machine learning, deep learning techniques have also been explored to capture complex temporal and behavioral patterns in blockchain transactions. However, these models often require large computational resources and lack interpretability, making them less practical for real-time fraud detection systems.

Graph-based approaches have gained attention due to the network nature of blockchain transactions. These methods model transactions as graphs where nodes represent entities and edges represent transaction flows. Techniques such as graph clustering and graph neural networks help identify suspicious clusters and coordinated fraudulent activities such as money laundering and wash trading.

Anomaly detection techniques are also used to identify transactions that deviate significantly from normal behavior. These methods are particularly useful in scenarios where labeled fraud data is limited. However, they often result in high false positive rates and require further validation mechanisms.

2.2 Limitations of Existing Methods

Despite the effectiveness of existing approaches, several limitations restrict their practical applicability in decentralized environments. Most traditional fraud detection systems rely on centralized data collection, where transaction data from multiple sources is aggregated

into a single repository for model training. This approach raises significant concerns regarding data privacy, security, and regulatory compliance.

Furthermore, financial institutions and decentralized platforms are often unwilling to share sensitive transaction data due to competitive and legal constraints. This results in limited datasets for model training, reducing the ability of models to generalize across diverse transaction patterns.

A further challenge lies in the ever-changing nature of fraudulent behavior. As attackers constantly refine their tactics, models that remain static often fail to keep up. Moreover, these systems frequently face issues with imbalanced data, since fraudulent cases make up only a small portion of the dataset, which can result in skewed or less accurate predictions.

2.3 Research Gap

The analysis of existing literature highlights a significant research gap in the development of privacy-preserving fraud detection systems for decentralized environments. While current models achieve high accuracy under centralized settings, they fail to address the challenge of secure collaboration among multiple organizations without sharing raw data.

There is a clear need for a framework that enables distributed model training while maintaining data confidentiality and complying with privacy regulations. Existing studies have limited focus on integrating machine learning with decentralized data environments in a scalable and efficient manner.

To bridge this gap, this study introduces a federated learning-based method that enables multiple participants to jointly develop a fraud detection model while exchanging only model parameters rather than actual transaction data. This approach maintains data confidentiality and enhances model performance by utilizing insights from distributed datasets.

3. PROPOSED METHODOLOGY

3.1 System Overview

The proposed system is structured as a privacy-focused fraud detection framework built on federated learning principles. Rather than storing all transaction data in a single location, the data is kept across multiple client nodes, each representing a separate organization. These clients train their own machine learning models using local data and transmit only the model parameters to a central server. The server then combines these updates to generate a global model, which is shared back with the clients for continued training. This cycle repeats over several communication rounds, supporting collaborative model improvement while keeping sensitive transaction data decentralized and protected.

The central server aggregates these parameters to form a global model, which is then redistributed to clients for further training. This iterative process continues for multiple communication rounds, enabling collaborative learning while ensuring that raw transaction data remains local and secure.

3.2 Data Preprocessing

The raw decentralized exchange (DEX) transaction dataset is preprocessed to ensure data quality and consistency before model training. Missing values in critical numerical fields such as transaction amounts and swap rates are handled by removing incomplete records.

Categorical features such as transaction type, token names, and exchange platforms are encoded using label encoding techniques to convert them into numerical representations. Numerical features are normalized using standardization to ensure uniform scaling across all input variables.

Time-based attributes are extracted from transaction timestamps, including hour of transaction and day of the week, which provide important behavioral insights.

3.3 Feature Engineering

Feature engineering is performed to extract meaningful patterns that help in identifying fraudulent transactions. Key features include transaction amount ratios, statistical deviations, and temporal indicators.

3.3.1 Swap Rate Deviation (Z-Score):

$$Z = \sigma |x - \mu| \quad (1)$$

Where:

- x = transaction swap rate
- μ = mean swap rate
- σ = standard deviation

This measure helps identify extreme deviations in swap rates, which may indicate price manipulation or abnormal trading behavior.

3.3.2 Amount Ratio Feature:

$$Ratio = AmountOut + \epsilon AmountIn \quad (2)$$

This ratio highlights abnormal token exchange behavior and prevents division errors using a small constant ϵ .

3.3.3 High-Value Transaction Flags: Transactions exceeding the 95th percentile of transaction amounts are flagged as high-value transactions. These flags are useful in detecting large-volume suspicious activities such as wash trading.

3.3.4 Temporal Features:

Table 1. Engineered Features Description

Feature Name	Description
SwapRateZScore	Measures deviation in swap rate
AmountRatio	Ratio of input to output tokens
HighAmountOut	High outgoing transaction flag
IsWeekhour	Late-night transaction indicator

Transactions occurring during late-night hours (00:00–05:00) are marked using a binary indicator. Such transactions often correlate with suspicious trading behavior.

3.4 Fraud Detection Model

The fraud detection task is formulated as a binary classification problem, where each transaction is labeled as either fraudulent or legitimate. A Logistic Regression model is chosen for its straightforward nature, ease of interpretation, and ability to efficiently process large volumes of data.

The model learns relationships between engineered features and transaction labels, enabling it to predict the likelihood of fraud for unseen data.

3.5 Federated Learning Process

The federated learning workflow consists of several communication rounds between the client nodes and the central server. In each round, every client trains a local model on its private data and then transmits the resulting model parameters to the server.

The server aggregates these parameters using the Federated Averaging algorithm to produce a global model.

3.5.1 Federated Averaging (FedAvg):

$$W_{global} = \sum_{k=1}^K \frac{n_k}{N} W_k \quad (3)$$

Where:

- K = number of clients
- n_k = data samples at client k
- N = total samples
- W_k = local model weights

This ensures that clients with larger datasets contribute proportionally more to the global model.

3.6 Summary of Methodology

The proposed methodology integrates data preprocessing, feature engineering, and federated learning to build a privacy-preserving fraud detection system. By combining statistical analysis with distributed model training, the system ensures accurate fraud detection while maintaining data confidentiality across multiple organizations.

4. IMPLEMENTATION AND RESULTS

4.1 Implementation Details

The system is developed in Python, combining data preprocessing, machine learning, and web-based visualization modules. The backend is built using the Flask framework, which offers RESTful APIs to handle model training and support real-time predictions.

The system processes a real-world decentralized exchange (DEX) dataset consisting of approximately 119,000 transactions. Data preprocessing, feature engineering, and model training are performed using libraries such as Pandas, NumPy, and Scikit-learn.

The federated learning process is simulated within a single execution environment, where multiple client nodes are represented logically rather than physically distributed systems. Each client trains a local Logistic Regression model, and the central server aggregates model parameters using the Federated Averaging algorithm.

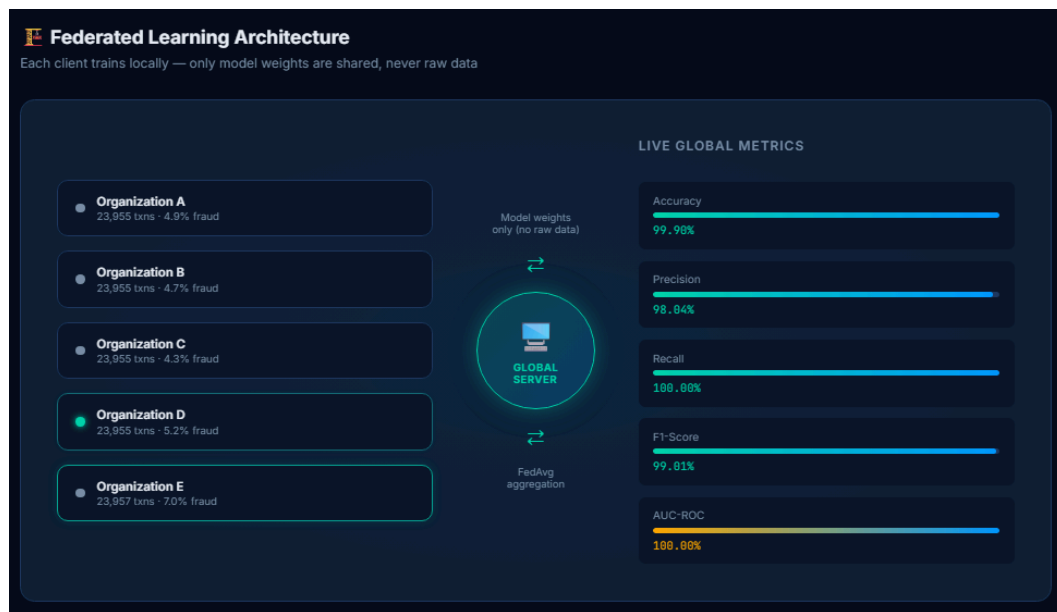


Figure 1. System Dashboard Interface

4.2 Experimental Setup

The experimental setup consists of five simulated client nodes, each representing an independent organization with its own dataset partition. The dataset is divided into non-identically distributed (non-IID) subsets to reflect real-world variations across different organizations.

The federated learning process is executed over 10 communication rounds. In each round, local models are trained independently and aggregated to form a global model.

The Logistic Regression model is configured with the following parameters:

- Maximum iterations: 500
- Random state: 42

Table 2. Client Data Distribution

Organization	Total Samples	Fraud Cases	Fraud Rate
Client A	23,955	1,169	4.88%
Client B	23,955	1,132	4.73%
Client C	23,955	1,032	4.31%
Client D	23,955	1,255	5.24%
Client E	23,957	1,669	6.97%

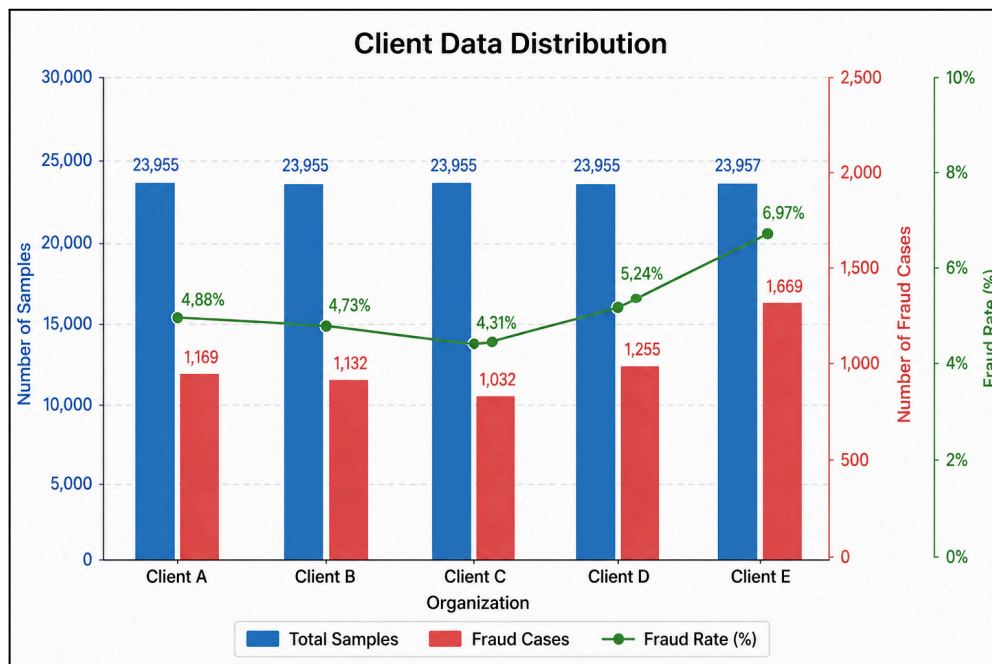


Figure 2. Data Distribution of Clients

4.3 System Functionality

The system provides two primary functionalities: federated model training and real-time transaction prediction.

The training process is executed through a web interface, where users can initiate live training. The backend streams real-time updates using Server-Sent Events (SSE), allowing visualization of training progress, including accuracy and performance metrics across rounds.

The prediction module allows users to input transaction details and receive instant classification results, including fraud probability and risk factors. This enhances interpretability and usability of the system.

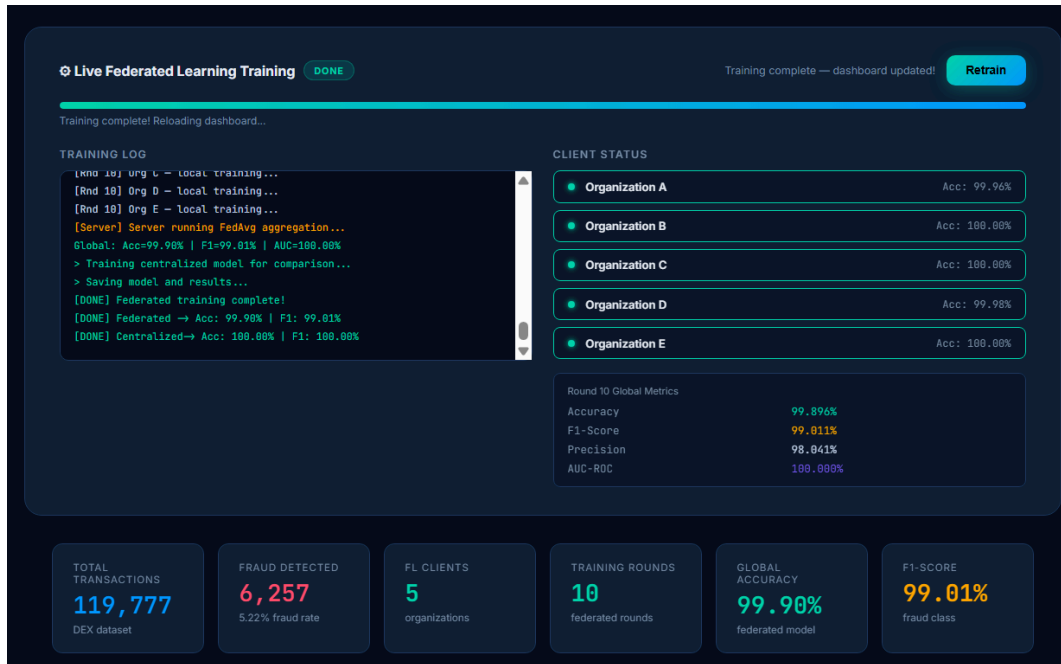


Figure 3. Live Training Process Visualization

4.4 Performance Evaluation

The effectiveness of the proposed system is assessed using common classification metrics such as accuracy, precision, recall, and F1-score. Together, these measures offer a well-rounded evaluation of the model's capability to accurately detect fraudulent transactions.

Table 3. Engineered Features Description

Metric	Federated Model	Centralized Model
Accuracy	99.89%	100.0%
Precision	98.04%	100.0%
Recall	100.0%	100.0%
F1-Score	99.01%	100.0%
AUC	99.99%	100.0%

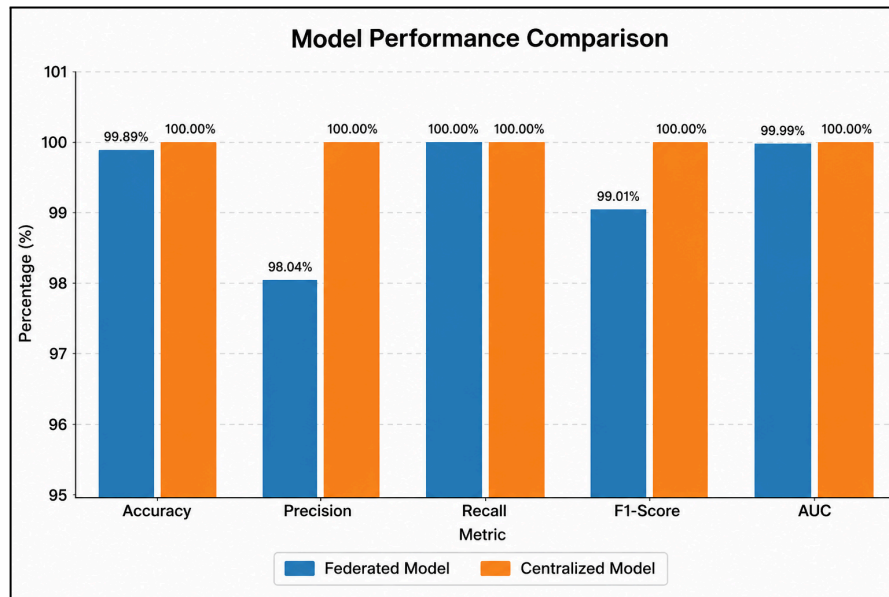


Figure 4. Accuracy and F1-Score Across Communication Rounds

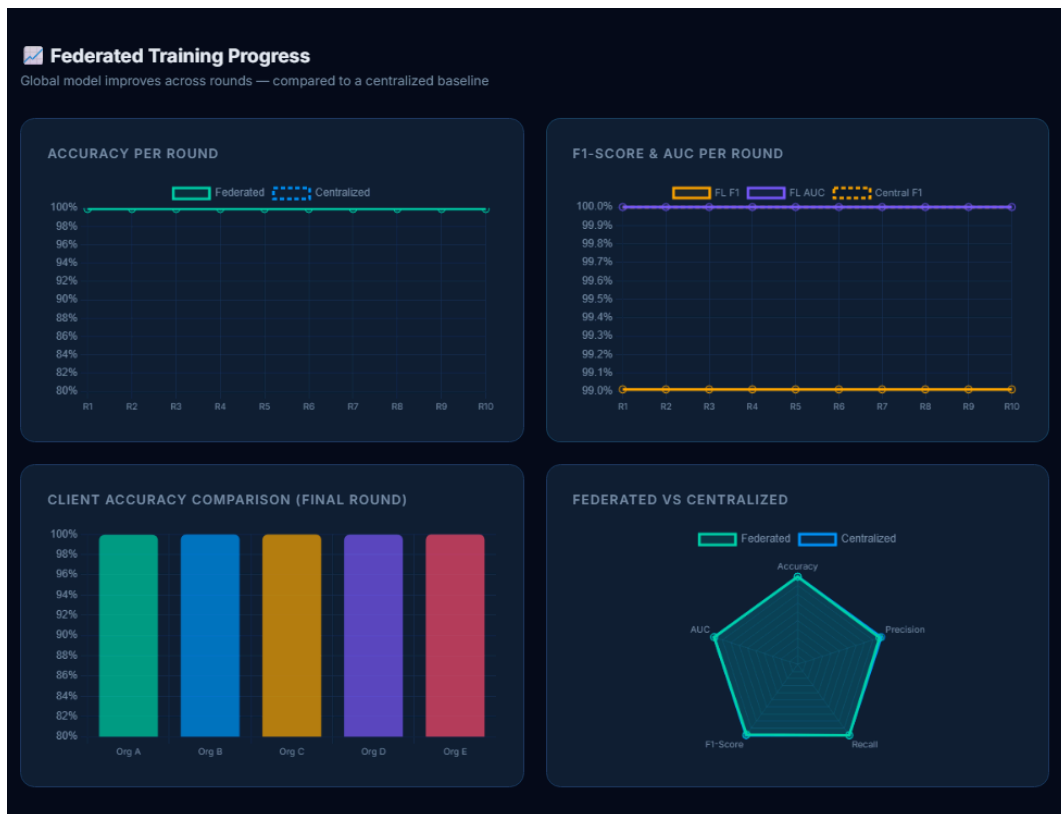


Figure 5. Federated Training Process

4.5 Results and Analysis

The experimental results demonstrate that the federated learning model achieves performance comparable to a centralized model, with minimal loss in accuracy. The global model converges rapidly within a limited number of communication rounds, indicating efficient learning across distributed datasets.

The high accuracy and F1-score indicate that the model effectively captures fraud patterns derived from feature engineering techniques. Additionally, the results confirm that federated learning can maintain strong predictive performance while preserving data privacy.

4.6 Limitations

Despite strong performance, the system has certain limitations. The federated learning setup is simulated within a single environment rather than implemented across physically distributed systems.

Additionally, the fraud labels used in the dataset are generated using heuristic rules rather than real-world ground truth, which may result in inflated performance metrics. The use of Logistic Regression limits the ability to capture highly complex patterns present in real-world fraud scenarios.

4.7 Summary of Implementation

This section outlined the implementation approach, experimental configuration, and performance analysis of the proposed system. The findings confirm that federated learning is a practical solution for privacy-focused fraud detection, delivering strong accuracy while ensuring data remains secure.

5. CONCLUSION

This research introduces a privacy-focused framework for detecting cryptocurrency fraud through federated learning. The system allows multiple decentralized participants to jointly train a machine learning model without exchanging sensitive transaction information, effectively tackling concerns related to data privacy and regulatory limitations within decentralized finance ecosystems.

The implementation integrates data preprocessing, feature engineering, and a Logistic Regression model within a federated learning setup. Experimental results demonstrate that the federated model achieves performance comparable to centralized approaches, with high accuracy and F1-score, while maintaining data confidentiality. The system also provides real-time visualization and prediction capabilities through an interactive web-based dashboard.

However, the current implementation is based on a simulated federated environment and utilizes heuristic-based fraud labeling, which may not fully represent real-world scenarios. Future work can focus on deploying the system across distributed infrastructures, incorporating real-world labeled datasets, and exploring advanced deep learning techniques for improved fraud detection performance.

6. FUTURE WORK

Future enhancements of the proposed system may include the implementation of a fully distributed federated learning architecture across multiple physical nodes. The integration of advanced models such as deep neural networks and graph-based learning techniques can further improve detection accuracy. Additionally, incorporating dynamic thresholding

mechanisms and real-time streaming data can enhance the adaptability of the system to evolving fraud patterns in decentralized financial ecosystems.

Acknowledgments

The authors would like to express their sincere gratitude to **Prof. Sweta Kahurke** for her continuous guidance, valuable suggestions, and constant support throughout the development of this research work. Her insights greatly contributed to shaping the direction and quality of this study.

We also extend our thanks to the Department of Artificial Intelligence, **JD College of Engineering and Management, Nagpur**, for providing the necessary resources and environment to successfully carry out this project.

Finally, we would like to acknowledge the support of our peers and all individuals who directly or indirectly contributed to the completion of this work.

REFERENCES

- [1] H. B. McMahan, E. Moore, D. Ramage, S. Hampson and B. A. y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data", *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, (2017), pp. 1273–1282.
- [2] Q. Yang, Y. Liu, T. Chen and Y. Tong, "Federated Machine Learning: Concept and Applications", *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 2, (2019), pp. 1–19.
- [3] V. Buterin, "A Next-Generation Smart Contract and Decentralized Application Platform", *Ethereum White Paper*, (2014).
- [4] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System", (2008).
- [5] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System", *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (2016), pp. 785–794.
- [6] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python", *Journal of Machine Learning Research*, vol. 12, (2011), pp. 2825–2830.
- [7] J. Brownlee, "Machine Learning Algorithms From Scratch", *Machine Learning Mastery*, (2016).
- [8] K. Zhao, S. Li, and Y. Wu, "Detecting Cryptocurrency Fraud Using Machine Learning Techniques", *IEEE Access*, vol. 8, (2020), pp. 148567–148576.
- [9] A. Singh and S. Jain, "Blockchain-Based Fraud Detection Systems: A Survey", *International Journal of Computer Applications*, vol. 182, no. 25, (2018), pp. 1–5.

- [10] D. Ron and A. Shamir, “Quantitative Analysis of the Full Bitcoin Transaction Graph”, *Financial Cryptography and Data Security*, (2013), pp. 6–24.
- [11] I. Goodfellow, Y. Bengio and A. Courville, “Deep Learning”, MIT Press, (2016).