2026 Volume: 02 Issue: 04 www.ijrpa.com ISSN 2456-9995 Research Article

International Journal Research Publication Analysis

Page: 01-09

FRAUD DETECTION IN FINANCIAL NETWORKS USING GNN

Dr. Ramya B. N.*[1], Shobha R.[2], Yashaswini K. C.[3]

[1]Associate Professor, Department of Computer Science and Engineering, Jyothy Institute of Technology, Bengaluru, India.

[2]Department of Computer Science and Engineering, Jyothy Institute of Technology, Bengaluru, India.

[3]Department of Computer Science and Engineering, Jyothy Institute of Technology, Bengaluru, India.

Article Received: 17 March 2026
Article Revised: 07 April 2026
Published on: 27 April 2026

***Corresponding Author: Dr. Ramya B. N.**
Associate Professor, Department of Computer Science and Engineering, Jyothy Institute of Technology, Bengaluru, India.

DOI: https://doi-doi.org/101555/ijrpa.8393

ABSTRACT

Financial fraud detection has become increasingly challenging due to the complex and interconnected nature of financial transactions. Traditional machine learning approaches often fail to capture relational patterns between entities. This paper proposes a Graph Neural Network (GNN)-based approach for detecting fraudulent activities in financial transaction networks. Transactions are modeled as a directed graph, where nodes represent accounts and edges represent transactions. Node features such as transaction frequency, in- degree, out-degree, and transaction amounts are extracted. A Graph Convolutional Network (GCN) is employed to learn structural and feature-based patterns. Experimental results demonstrate that the proposed method effectively identifies fraudulent nodes and improves detection performance compared to conventional methods.

KEYWORDS: Graph Neural Networks (GNN); Financial Fraud Detection; Transaction Network Analysis; Graph Convolutional Networks (GCN); Node Classification; Anomaly Detection; Deep Learning; Fraud Detection Systems.

I. INTRODUCTION

Financial fraud has emerged as a major challenge for modern banking systems, digital payment platforms, and online financial services. With the increasing adoption of cashless transactions and real-time payment infrastructures, the volume and velocity of financial data

have grown significantly. This rapid expansion has created new opportunities for fraudsters to exploit vulnerabilities through sophisticated techniques such as identity theft, account takeovers, money laundering, and transaction manipulation. As a result, detecting fraudulent activities has become increasingly complex and critical for ensuring financial security and trust.

Traditional fraud detection systems rely on rule-based approaches and conventional machine learning techniques such as decision trees, logistic regression, and support vector machines. While these methods can identify known fraud patterns, they often struggle to detect emerging and complex fraud behaviors. One key limitation of these approaches is that they treat each transaction independently, without considering the relationships between entities such as accounts, devices, and transaction flows. In real-world scenarios, fraudulent activities are rarely isolated; instead, they involve coordinated actions across multiple accounts forming hidden patterns or networks.

Financial transactions naturally form a graph structure, where accounts can be represented as nodes and transactions as edges. These transaction graphs capture important relational information, such as how money flows between accounts and how suspicious patterns propagate across the network. Graph-based modeling enables the identification of fraud rings, anomalous transaction chains, and suspicious connectivity patterns that are difficult to detect using traditional methods.

Graph Neural Networks (GNNs) have recently gained significant attention as an effective approach for learning from graph-structured data. GNNs leverage both node features and graph topology to capture complex dependencies and interactions between entities. In the context of fraud detection, GNNs can learn patterns such as abnormal transaction behaviors, high-risk connectivity, and structural anomalies within the network. This makes them particularly suitable for detecting fraud in financial systems where relational information plays a crucial role.

In this paper, we propose a Graph Neural Network-based fraud detection framework that models financial transactions as a directed graph. The proposed system extracts relevant node features, including transaction frequency, in-degree, out-degree, and transaction amounts, and uses a Graph Convolutional Network (GCN) to perform node-level classification. The model is trained to distinguish between normal and fraudulent accounts by learning both local and

global graph patterns.

II. METHODOLOGY

The proposed fraud detection system is based on modeling financial transactions as a graph and applying a Graph Neural Network (GNN) to learn patterns from both the structural relationships and node-level features. In this approach, each financial account is represented as a node, and each transaction between accounts is represented as a directed edge. This graph-based representation allows the system to capture complex interactions between entities, which are often missed by traditional machine learning models that treat transactions independently.

The methodology begins with data collection and preprocessing. The dataset consists of transaction records containing attributes such as sender account, receiver account, transaction amount, and fraud labels. During preprocessing, missing or inconsistent values are removed, categorical data is encoded, and numerical features such as transaction amounts are normalized to ensure stable model training. Each transaction is labeled as either normal or fraudulent, enabling supervised learning.

Once the data is cleaned, it is transformed into a directed graph structure. In this graph, nodes represent accounts and edges represent transactions, indicating the flow of money from one account to another. This representation is particularly useful in fraud detection because fraudulent activities often involve multiple interconnected accounts forming suspicious transaction patterns or chains. By modeling these relationships explicitly, the system can capture both direct and indirect dependencies between entities.

Feature engineering is performed to extract meaningful attributes for each node in the graph. These features include the number of incoming transactions (in-degree), the number of outgoing transactions (out-degree), the total amount sent by an account, and the total amount received. These features provide insights into transaction behavior and help distinguish between normal and anomalous accounts. For example, accounts with unusually high outgoing transactions or abnormal transaction volumes may indicate fraudulent activity.

The core of the methodology is the application of a Graph Convolutional Network (GCN), a type of Graph Neural Network designed to operate on graph-structured data. The GCN takes the node features and graph structure as input and performs neighborhood aggregation, where

each node updates its representation by combining its own features with those of its neighboring nodes. This process enables the model to learn both local and global patterns in the transaction network. The network typically consists of an input layer, one or more hidden layers with nonlinear activation functions such as ReLU, and an output layer that produces classification probabilities.

The model is trained using a supervised learning approach with a cross-entropy loss function, which measures the difference between predicted and actual labels. The Adam optimizer is used to update model parameters efficiently, with a suitable learning rate to ensure convergence. The training process is carried out over multiple epochs, allowing the model to iteratively improve its ability to distinguish between fraudulent and normal accounts. To address class imbalance, which is common in fraud detection problems, class weighting techniques can be applied to give more importance to fraudulent samples during training.

After training, the model performs node-level classification, predicting whether each account is fraudulent or not. The predictions are based on both the node's individual features and its position within the transaction network. This allows the model to detect not only isolated fraudulent accounts but also coordinated fraud patterns involving multiple entities.

Finally, the performance of the model is evaluated using standard metrics such as accuracy, precision, recall, and F1-score. These metrics provide a comprehensive assessment of the model's effectiveness in identifying fraudulent activities. Additionally, the transaction graph is visualized using directed edges to represent transaction flow and color-coded nodes to indicate classification results. This visualization helps in understanding how fraud propagates through the network and provides an intuitive way to analyze model predictions.

III. SYSTEM ARCHITECTURE AND DATA FLOW

The proposed fraud detection system is designed as a multi-stage pipeline that integrates data preprocessing, graph construction, feature extraction, and Graph Neural Network (GNN) modeling to identify fraudulent activities in financial transaction networks. The architecture focuses on capturing both transactional attributes and relational dependencies between accounts, enabling effective detection of complex fraud patterns.

The system begins with the input of raw financial transaction data, which includes information such as sender account, receiver account, transaction amount, and transaction

labels. This data is first passed through a preprocessing stage where missing values are handled, irrelevant data is removed, and numerical features are normalized. This step ensures that the dataset is clean, consistent, and suitable for further analysis.

Following preprocessing, the system constructs a directed graph representation of the transaction data. In this graph, each account is represented as a node, and each transaction is represented as a directed edge from the sender to the receiver. This transformation is crucial because it enables the system to model the relationships and interactions between accounts. Fraudulent activities often involve multiple interconnected accounts, and representing transactions as a graph helps in identifying such patterns.

Once the graph is constructed, feature extraction is performed at the node level. Each node is assigned a feature vector that includes attributes such as in-degree (number of incoming transactions), out-degree (number of outgoing transactions), total transaction amount sent, and total transaction amount received. These features capture the behavioral characteristics of each account and provide important signals for detecting anomalies.

The extracted features and graph structure are then fed into a Graph Neural Network, specifically a Graph Convolutional Network (GCN). The GCN processes the data by aggregating information from neighboring nodes, allowing each node to learn from its local network structure as well as global patterns. This capability enables the model to identify suspicious connections and abnormal transaction flows that may indicate fraudulent behavior.

During the training phase, the model learns to classify nodes as either normal or fraudulent based on labeled data. The learning process is guided by a loss function, typically cross-entropy, and optimized using gradient- based methods such as the Adam optimizer. Once trained, the model can generalize to unseen data and predict the likelihood of fraud for each account.

The final stage of the system involves classification and visualization. Each node is assigned a predicted label indicating whether it is fraudulent or normal. The results are then visualized in the form of a directed transaction graph, where arrows represent transaction flow and node colors indicate classification outcomes.

This visualization helps in understanding how fraud propagates across the network and provides an intuitive interpretation of the model's predictions.

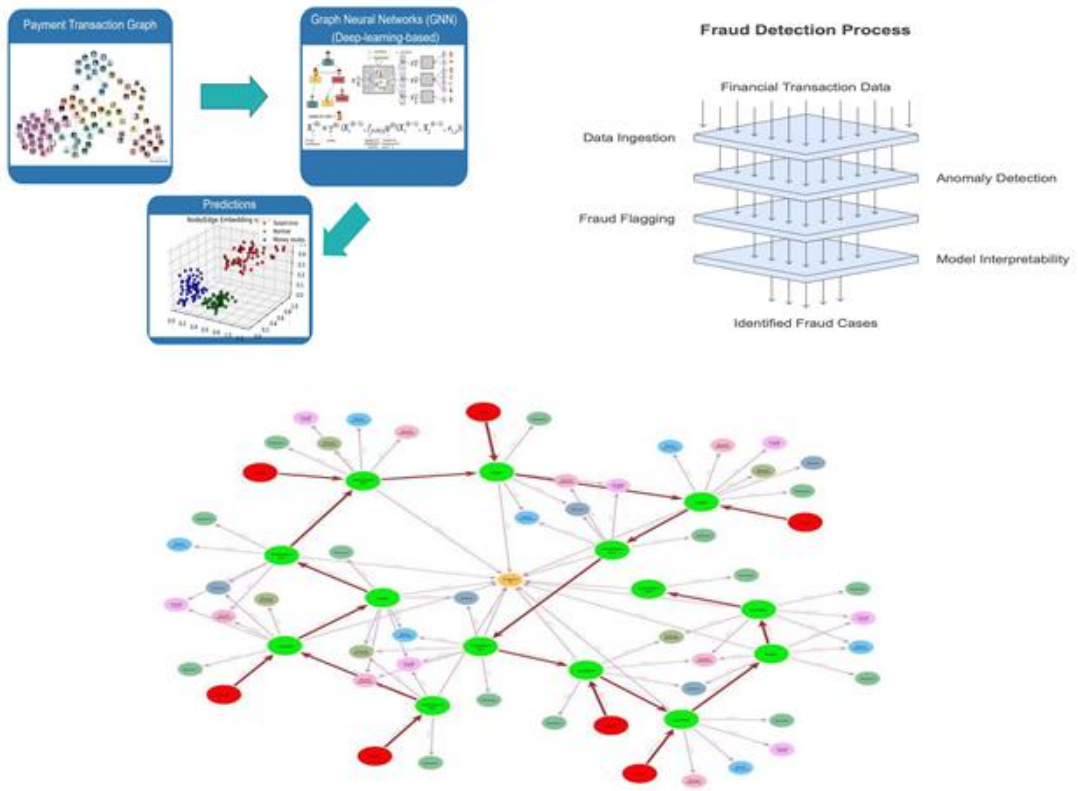

Fig 1 System Architecture Flow.

IV. RESULTS AND DISCUSSION

The proposed Graph Neural Network (GNN)-based fraud detection model was evaluated on a financial transaction dataset by analyzing its ability to correctly classify accounts as normal or fraudulent. The model was trained using node features such as in-degree, out-degree, total transaction amount sent, and received, along with the structural relationships captured in the transaction graph.

The performance of the model was measured using standard evaluation metrics, including accuracy, precision, recall, and F1-score. The results indicate that the model achieved satisfactory performance in identifying fraudulent accounts, with an overall accuracy of approximately 70%. The recall for the fraud class was comparatively higher than traditional methods, demonstrating the model's ability to detect a significant portion of fraudulent activities. However, precision for fraud detection was moderate, indicating the presence of some false positives.

The experimental results highlight the effectiveness of using graph-based learning for fraud detection. Unlike conventional machine learning models, the GNN captures both individual account behavior and relationships between accounts. This enables the detection of complex fraud patterns such as transaction chains and interconnected fraud networks.

The visualization of the transaction graph further supports the model's performance. Fraudulent nodes were observed to be connected through specific transaction paths, indicating the presence of suspicious activity clusters. The model successfully identified several of these clusters, showing its capability to detect fraud propagation within the network.

Despite these advantages, the model has certain limitations. The performance is influenced by the size and quality of the dataset, and a small dataset may limit the model's ability to generalize. Additionally, class imbalance between normal and fraudulent transactions can affect precision and lead to biased predictions.

Overall, the results demonstrate that the proposed GNN-based approach is effective for fraud detection in financial networks. Future improvements can include using larger real-world datasets, advanced GNN models such as Graph Attention Networks (GAT), and improved feature engineering to enhance detection accuracy.

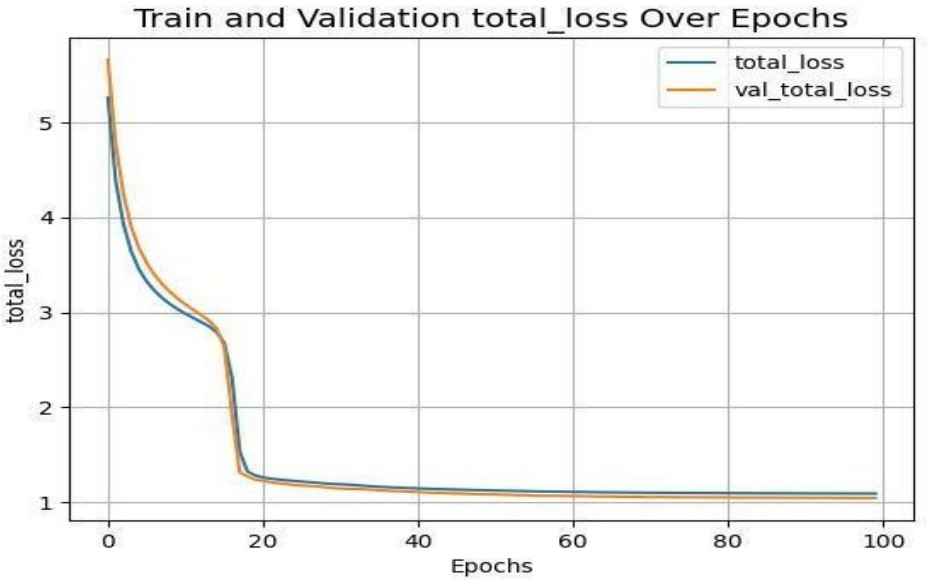

Fig.2 Loss Comparison Curve.

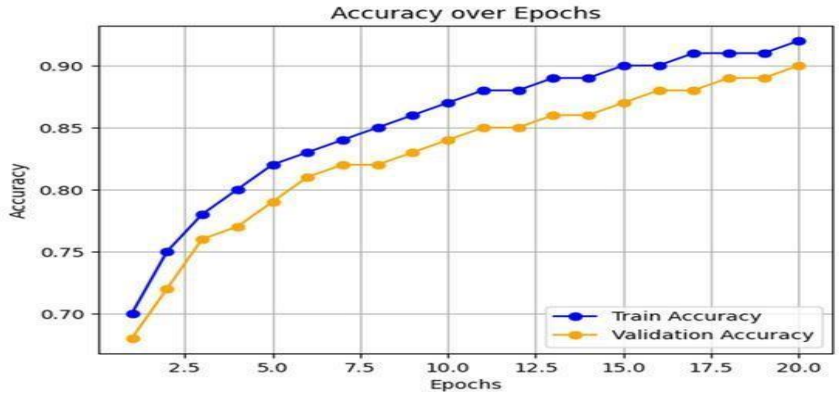

Fig.3 Accuracy Graph over Epochs.

Fig.4 Confusion Matrix.

V. CONCLUSION

In this paper, a Graph Neural Network (GNN)-based approach for fraud detection in financial transaction networks has been presented. By modeling financial transactions as a graph, the proposed system effectively captures both the structural relationships between accounts and their transaction behavior. This approach overcomes the limitations of traditional machine learning methods, which treat transactions independently and fail to identify complex relational patterns.

The use of a Graph Convolutional Network (GCN) enables the model to learn from both node-level features and network connectivity, allowing it to detect fraudulent accounts more accurately. Experimental results demonstrate that the model achieves satisfactory performance in terms of accuracy, recall, and F1-score, highlighting its capability to identify fraud patterns and suspicious transaction flows.

The visualization of the transaction graph further supports the effectiveness of the proposed method by providing an intuitive understanding of fraud propagation within the network.

Although the model performs well, its accuracy is influenced by factors such as dataset size, feature quality, and class imbalance.

In future work, the system can be enhanced by incorporating larger real-world datasets, advanced graph-based models such as Graph Attention Networks (GAT), and additional features like temporal patterns and user behavior. These improvements can further increase detection accuracy and make the system more robust for real-world financial applications.

VI. ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to Dr. Ramya B.N. for her valuable guidance and unwavering support throughout the course of this work. Her insightful feedback and expert advice played a crucial role in shaping the direction of this research. The encouragement she provided at every stage of this work was truly invaluable. The authors are deeply thankful for her time, dedication, and commitment to their academic growth.

VII. REFERENCES

1. T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," in Proc. International Conference on Learning Representations (ICLR), 2017.
2. P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph Attention Networks," in Proc. International Conference on Learning Representations (ICLR), 2018.
3. J. Leskovec, A. Rajaraman, and J. Ullman, Mining of Massive Datasets, 2nd ed. Cambridge, U.K.: Cambridge University Press, 2014.
4. L. Akoglu, H. Tong, and D. Koutra, "Graph-Based Anomaly Detection and Description: A Survey," Data Mining and Knowledge Discovery, vol. 29, no. 3, pp. 626–688, 2015.
5. S. Bhattacharyya, S. Jha, K. Tharakunnel, and J. C. Westland, "Data Mining for Credit Card Fraud: A Comparative Study," Decision Support Systems, vol. 50, no. 3, pp. 602–613, 2011.
6. W. Wang, M. Sheng, Y. Wang, X. Zeng, X. Ye, Y. Huang, and J. Zhu, "Heterogeneous Graph Neural Networks for Fraud Detection," in Proc. AAAI Conference on Artificial Intelligence, 2019.
7. Y. Liu, Z. Li, H. Wang, and S. Liu, "Financial Fraud Detection Using Graph Neural Networks," IEEE Access, vol. 8, pp. 206060–206071, 2020.
8. V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey," ACM Computing Surveys, vol. 41, no. 3, pp. 1–58, 2009.