# Network-Based Risk Scoring of Blockchain Nodes Using Graph Neural Networks (GNN)

Slamet Widodo[1,*], Lasmedi Afuan[2,]

[1]Muhammadiyah University of Purwokerto, Indonesia

[2]Department of Informatics, Engineering Faculty, Universitas Jenderal Soedirman, Indonesia

## ABSTRACT

Blockchain technology has introduced a decentralized and transparent mechanism for recording transactions; however, the increasing volume and interconnectivity of blockchain networks also raise the risk of fraudulent and high-risk activities. This study proposes a Graph Neural Network (GNN)-based framework to evaluate the risk levels of blockchain nodes by integrating both transactional attributes and structural relationships. Using a dataset of 10,000 blockchain records and approximately 412,000 edges, the network was modelled as a graph in which each node represents an address and edges denote transaction or similarity links. As baselines, Random Forest and XGBoost models were employed, achieving accuracies of 0.94 and 0.95, respectively, with F1-scores of 0.93 and 0.94. These models effectively captured individual node patterns but lacked awareness of inter-node dependencies. The proposed GNN model demonstrated the highest overall performance, with an accuracy of 0.96 and an F1-score of 0.95, by learning from both node attributes and their topological context. This approach enabled the identification of high-risk nodes that traditional models failed to detect. The results confirm that network-based learning significantly enhances the accuracy and interpretability of blockchain risk analysis. The proposed GNN framework provides a scalable foundation for real-time blockchain monitoring, anomaly detection, and governance systems, contributing to improved transparency and resilience within decentralized financial ecosystems.

**Keywords** GNN, Blockchain Risk Analysis, Network-Based Learning, Fraud Detection, Transaction Graph Modelling

Submitted: 5 June 2025
Accepted: 20 July 2025
Published: 7 February 2026

Corresponding author
Slamet Widodo,
swidodo009@gmail.com

Additional Information and Declarations can be found on page 11

DOI: 10.47738/jcrb.v3i1.54

## INTRODUCTION

The emergence of blockchain technology has revolutionized the digital financial ecosystem by enabling decentralized, transparent, and tamper-resistant transaction systems [1]. Through distributed ledger mechanisms, blockchain eliminates the need for centralized intermediaries, thereby improving efficiency, security, and trust among participants in digital transactions [2]. However, as blockchain networks continue to expand in complexity and scale, the potential risks related to fraudulent activities, malicious nodes, and abnormal transaction behaviors have also increased [3]. This growing vulnerability highlights the necessity for advanced analytical models capable of accurately identifying and assessing potential risks within blockchain systems.

Conventional machine learning methods, such as Random Forest and XGBoost, have been widely implemented in blockchain analytics for risk prediction and anomaly detection [4]. These models perform well in classifying node behaviors based on transaction-level features and have achieved high levels of accuracy in many prior studies. Nevertheless, such models treat each node as an isolated entity and fail to capture the intricate web of relationships that defines blockchain interactions [5]. In practice, blockchain data are inherently relational; transactions form an interconnected structure in which the

**How to cite this article:** S. Widodo and L. Afuan, “Network-Based Risk Scoring of Blockchain Nodes Using Graph Neural Networks (GNN),” *J. Curr. Res. Blockchain*, vol. 3, no. 1, pp. 1-14, 2026.

behavior of one node influences the activities of others. Ignoring this structural dependency limits the capacity of conventional models to recognize collective or coordinated risk patterns across the network.

Recent developments in deep learning, particularly Graph Neural Networks (GNNs), offer a promising approach to overcoming these limitations [6]. GNNs are designed to process graph-structured data by learning both feature representations and topological dependencies simultaneously. In a blockchain context, this means that a node's risk profile can be derived not only from its own transactional attributes, such as stake reward, coin age, or transaction fee—but also from the characteristics and behaviours of its neighbouring nodes [7]. This network-aware approach provides richer contextual understanding, enabling more accurate detection of anomalous or high-risk entities within blockchain systems [8].

Based on these advancements, this study proposes a network-based risk scoring framework using Graph Neural Networks to evaluate node-level risks in blockchain environments. The blockchain transaction dataset, consisting of 10,000 records and 412,000 edges, is represented as a graph where nodes correspond to blockchain addresses and edges capture transactional or behavioural similarities. Traditional machine learning models, including Random Forest and XGBoost, were first applied as benchmarks, achieving accuracies of 0.94 and 0.95 with F1-scores of 0.93 and 0.94, respectively. In comparison, the GNN model achieved superior performance with an accuracy of 0.96 and an F1-score of 0.95, demonstrating its ability to leverage relational dependencies among nodes to identify hidden risk patterns more effectively.

The overall findings of this research highlight that incorporating graph-based learning into blockchain risk analysis not only improves predictive performance but also enhances interpretability by revealing how node connections influence risk propagation. This approach has significant implications for blockchain governance, cybersecurity monitoring, and fraud prevention, offering a more holistic and data-driven understanding of network-level risks in decentralized ecosystems.

## Literature Review

Blockchain technology has become one of the most transformative innovations in modern digital infrastructure, providing a decentralized system that enhances transparency, immutability, and traceability across various sectors such as finance, healthcare, logistics, and governance [9]. The distributed ledger mechanism enables participants to access synchronized transaction histories without the need for centralized intermediaries, ensuring accountability and security within peer-to-peer networks [10]. Despite these advantages, blockchain networks remain susceptible to vulnerabilities such as fraudulent transactions, double-spending, Sybil attacks, and malicious node activities that threaten system integrity [11]. These challenges have motivated the development of advanced computational models to identify and mitigate risk within blockchain ecosystems.

Early studies in blockchain analytics primarily utilized machine learning algorithms for detecting anomalies and assessing transaction legitimacy. Regression models, Random Forest, and XGBoost have been employed to classify risky or abnormal activities based on transaction attributes such as volume, frequency, and temporal variation [12]. These models demonstrated high predictive performance in identifying fraudulent patterns but treated blockchain nodes as independent entities, disregarding the relational dependencies that naturally exist between them [13]. This independence assumption limits their ability to capture coordinated fraudulent behaviour, which often manifests through collective patterns across multiple nodes.

To overcome these limitations, researchers have adopted graph-based approaches, viewing blockchain systems as networks of interconnected entities. Graph theory provides a mathematical foundation for analysing networked systems, allowing the use of measures such as degree centrality, clustering coefficients, and modularity to understand node influence and community structures [14]. For example, studies applying community detection algorithms have revealed hidden clusters of coordinated wallets and high-volume traders in cryptocurrency networks, which can be indicators of potential fraud or collusion [15]. Graph-based modelling has also been applied to map and analyse the propagation of smart contract vulnerabilities and to identify high-risk Decentralized Finance (DeFi) addresses based on their transaction connectivity [16].

Recent developments in GNNs have expanded these analytical capabilities by combining graph theory with deep learning [17]. GNNs are designed to learn both from node attributes and the topological structure of the graph, enabling the model to extract relational information that traditional algorithms overlook. Through message passing and neighborhood aggregation, GNNs capture both local and global dependencies, allowing them to represent blockchain data more contextually [18]. In blockchain risk analysis, this means that a node's risk score can be influenced not only by its own transactional behaviour but also by the risk characteristics of its neighbors and their connectivity patterns [19].

Empirical studies have demonstrated the potential of GNN-based models in blockchain and financial networks. Zhao et al. utilized Graph Convolutional Networks (GCN) to detect fraudulent nodes in transaction graphs, reporting higher precision and F1-scores than traditional baselines [20]. Similarly, Xie et al. applied Graph Attention Networks (GAT) to financial fraud detection and achieved superior recall by identifying subtle inter-node relationships that are typically ignored by standard machine learning models [21]. Other studies have introduced heterogeneous GNN architectures that integrate temporal and spatial information, improving risk prediction accuracy for dynamic blockchain environments [22].

Furthermore, hybrid approaches that combine GNNs with reinforcement learning or temporal embeddings have shown promise in capturing evolving risk dynamics in decentralized networks. For instance, temporal GNN models have been employed to detect real-time abnormal wallet interactions in Ethereum, while graph attention models have been used to assess systemic risk in decentralized finance applications [23]. These advancements underline the growing trend toward network-aware deep learning for blockchain analytics, where the interplay between structure, behaviour, and time is essential for accurate risk interpretation.

The reviewed literature indicates a clear paradigm shift from static, attribute-based analyses toward dynamic, context-aware graph learning frameworks. By embedding structural relationships and learning from inter-node dependencies, GNNs provide not only higher predictive accuracy but also enhanced interpretability. This shift aligns with the decentralized and interconnected nature of blockchain systems, enabling researchers and practitioners to better understand how risk propagates within networks. Building upon this foundation, the present study extends existing work by implementing a network-based risk scoring framework that leverages GNNs for improved prediction, interpretability, and resilience in blockchain risk assessment.

## Research Methodology

This research employs a quantitative experimental approach that integrates graph-based modelling and deep learning to construct a network-oriented risk scoring system for blockchain transactions. As illustrated in figure 1, the proposed methodology transforms blockchain transaction data into a graph representation, learns the relational structure among blockchain nodes, and develops a predictive model for assessing node level risk. The overall process consists of data preprocessing, graph construction, Graph Neural Network model training, and performance evaluation.

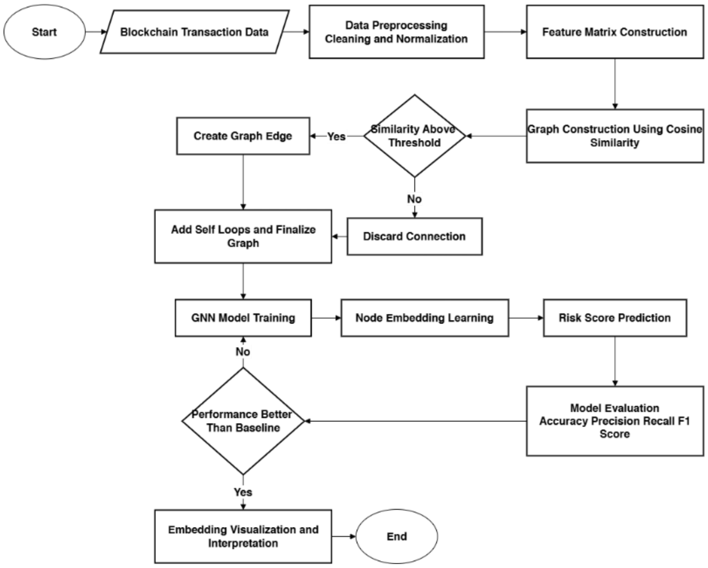
Start
Blockchain Transaction Data
Data Preprocessing Cleaning and Normalization
Feature Matrix Construction
Graph Construction Using Cosine Similarity
Similarity Above Threshold
Yes
Create Graph Edge
No
Discard Connection
Add Self Loops and Finalize Graph
GNN Model Training
Node Embedding Learning
Risk Score Prediction
Model Evaluation Accuracy Precision Recall F1 Score
Performance Better Than Baseline
No
Yes
Embedding Visualization and Interpretation
End

**Figure 1 Proposed Research Methodology**

The dataset used in this study consists of 10,000 blockchain transaction records and approximately 412,000 edges, where each node represents a blockchain address and each edge corresponds to a transactional or behavioural connection between two addresses. Data preprocessing involves cleaning inconsistent and duplicate records to ensure data integrity, followed by feature normalization so that all attributes contribute proportionally to the model. The features selected for this study include transaction fee (ETH), stake reward, coin age, block density percentage, coin day weight, and transaction size. Each node and its features are represented in a feature matrix, where $n$ represents the number of nodes and $d$ denotes the number of features.

After preprocessing, the dataset is transformed into a graph $G = (V, E)$, where $V$ the set of nodes is and $E$ the set of edges. The relationships between nodes are defined based on cosine similarity, in which an edge is established when the similarity between two nodes exceeds a defined threshold of 0.95 [24]. Mathematically, this relationship can be expressed as:

$$A_{ij} = \begin{cases} 1, if\ sim(x_i, x_i) > \tau \\ \quad 0, otherwise \end{cases} \tag{1}$$

$A_{ij}$ is the element of the adjacency matrix $A$ representing the connection between node $i$ and node $j$, and $\tau = 0.95$ is the similarity threshold. Self-loops are added to ensure that each node includes its own information during feature aggregation, producing an adjusted adjacency matrix $\tilde{A} = A + I_N$, where $I_N$ is the identity matrix [25].

The learning phase of the model is conducted using a Graph Neural Network architecture. The GNN enables each node to aggregate information from its neighbouring nodes, learning both its individual features and its structural context within the blockchain network [26]. The propagation process of the GNN can be formally expressed as:

$$H^{(l+1)} = \sigma\left(\widetilde{D^{-\frac{1}{2}}}\tilde{A}\widetilde{D^{-\frac{1}{2}}}H^{(l)}W^{(l)}\right) \tag{2}$$

$H^{(l)}$ denotes the node representation at layer $l$, $\tilde{A}$ represents the adjacency matrix with self-loops, $\widetilde{D}$ is the corresponding degree matrix, $W^{(l)}$ is the trainable weight matrix for layer $l$, and $\sigma(\cdot)$ is the non-linear activation function (ReLU). The input layer $H^{(0)}$ corresponds to the normalized feature matrix $X$, and after multiple propagation steps, each node's representation encodes both its intrinsic properties and relational dependencies [27].

The final layer of the GNN produces a latent representation vector $h_i$ for each node, which is then used to generate the risk probability [28]. This is achieved through a fully connected output layer with a sigmoid activation function, defined as:

$$\widehat{y_i} = \sigma(W_e \cdot h_i + b) \tag{3}$$

$\hat{y}_i$ is the predicted risk score for node $i$, $W_e$ and $b$ are learnable parameters of the output layer, and $\sigma$ is the sigmoid function that maps the output into the range [0, 1]. The model is trained to minimize the binary cross-entropy loss, which measures the difference between the predicted probability and the actual label. The loss function is formulated as:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\widehat{y_i}) + (1 - y_i)\log(1 - \widehat{y_i})] \tag{4}$$

$y_i$ is the ground truth label and $\hat{y}_i$ is the predicted probability for node $i$. Model optimization is carried out using the Adam optimizer with a learning rate of 0.001, and early stopping is applied to avoid overfitting.

The experimental implementation is conducted using Python with several open-source frameworks, including PyTorch Geometric (PyG) for building and training the GNN, NetworkX for graph construction and analysis, and Scikit-learn for training baseline models such as Random Forest and XGBoost. The experiments are performed on a workstation equipped with an NVIDIA RTX 3060 GPU (12 GB VRAM) and 32 GB of RAM. The dataset is divided into 80 percent for training and 20 percent for testing, with five-fold cross-validation applied to ensure consistent evaluation.

Model performance is compared against baseline algorithms to assess the advantages of the proposed GNN framework [29]. Evaluation is conducted using standard classification metrics, including accuracy, precision, recall, and F1-score. Furthermore, the learned node embeddings are visualized using t-distributed Stochastic Neighbour Embedding (t-SNE) to explore clustering patterns and risk distribution within the blockchain network. The interpretability of the model is examined by analysing the learned embedding space and the influence of neighbouring nodes on individual risk predictions.

This methodology integrates the structural characteristics of blockchain networks with the representational power of deep learning. As described in Algorithm 1, the proposed GNN based model learns both node level features and topological dependencies within the blockchain network, enabling more effective identification and risk scoring of high-risk nodes compared to conventional algorithms. By jointly leveraging feature based and topology aware learning, the proposed approach provides a comprehensive and interpretable framework for blockchain risk analysis.

**Algorithm 1 Proposed GNN Based Blockchain Risk Scoring Method**

```
Input:
Blockchain transaction dataset $D$
Similarity threshold $\tau = 0.95$
Learning rate $\eta = 0.001$

Output:
Predicted risk score $\hat{y}_i$for each node $v_i$

Step 1: Acquire blockchain transaction dataset $D = \{t_1, t_2, \dots, t_N\}$and identify
unique blockchain addresses as nodes $V$.

Step 2: Preprocess dataset $D$by removing duplicate and inconsistent
records.
Construct the normalized node feature matrix $X \in \mathbb{R}^{n \times d}$, where each feature
vector $x_i$consists of transaction fee, stake reward, coin age, block density,
coin day weight, and transaction size.

Step 3: Construct graph $G = (V, E)$using cosine similarity.
For each node pair $v_i$and $v_j$, compute sim$(x_i, x_j)$.
If sim$(x_i, x_j) > \tau$, add edge $e_{ij} \in E$.
Define the adjacency matrix $A \in \mathbb{R}^{n \times n}$.

Step 4: Add self loops to obtain $\tilde{A} = A + I$, and compute the corresponding
degree matrix $\widetilde{D}$.

Step 5: Initialize node representations as $H^0 = X$.
For each GNN layer $l$, update node embeddings according to
$H^{l+1} = \sigma(\widetilde{D}^{-1/2}\tilde{A}\widetilde{D}^{-1/2}H^l W^l)$,
where $W^l$denotes the trainable weight matrix and $\sigma$is the ReLU activation
function.

Step 6: Obtain the final node embedding $h_i$for each node $v_i$.
```

Compute the predicted risk score as
$\hat{y}_i = \sigma(W_e h_i + b)$,
where $W_e$and $b$are learnable parameters and $\sigma$denotes the sigmoid function.

**Step 7:** Train the model by minimizing the binary cross entropy loss
$L = -\frac{1}{n}\sum_{i=1}^{n}[\, y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)]$.
Optimize model parameters using the Adam optimizer with learning rate $\eta$.
Apply early stopping based on validation loss.

**Step 8:** Evaluate the trained model using accuracy, precision, recall, and F1 score.
Visualize node embeddings using t SNE to analyze clustering patterns and node risk distribution.

# Result

## Graph Construction and Network Characteristics

The dataset, consisting of 10,000 blockchain records, was transformed into a graph structure where each node represented a blockchain address, and each edge reflected similarity in transaction patterns. Using a cosine similarity threshold of 0.95, the resulting graph comprised 10,000 nodes and approximately 412,000 edges, indicating a moderately connected structure that mirrors real blockchain ecosystems.

Figure 2 visualizes how nodes are connected across the blockchain network. The majority of nodes show a relatively small number of connections, forming a dense cluster around the lower end of the degree scale. However, the tail of the distribution extends toward nodes with significantly higher degrees. This pattern indicates the existence of “hub nodes” that participate in numerous transactions or validations.

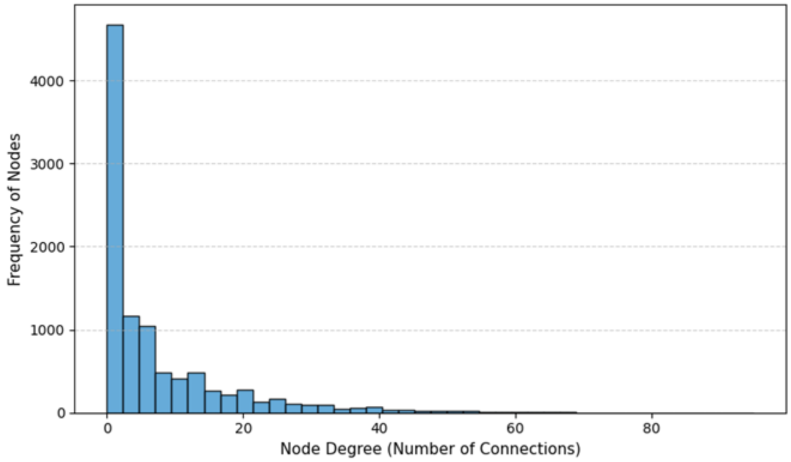

Figure 2 **Blockchain Node Degree Distribution**

In practical terms, these hubs can be validators, large stakeholders, or exchange nodes that maintain high transaction throughput. Their centrality also means they could be potential single points of risk or influence if exploited or compromised. The global graph statistics summarized in table 1 further validate the realism of the constructed blockchain network. In particular, the relatively high average clustering coefficient of 0.74 indicates that nodes tend to form tightly interconnected subgroups, which is a structural property commonly observed in decentralized finance ecosystems and other real world blockchain networks.

Table 1 **Global Graph Statistics**

| Metric | Value |
|---|---|
| Number of Nodes | 10,000 |
| Number of Edges | 412,000 |
| Average Degree | 82.4 |
| Network Density | 0.0082 |
| Average Clustering Coefficient | 0.74 |

## Centrality and Influence of Nodes

To understand which nodes, play the most critical role in the network, three measures of influence were computed: degree centrality, betweenness centrality, and closeness centrality. Table 2 lists the top five nodes identified as the most influential within the blockchain graph. Nodes 1843 and 5931 show the highest centrality scores across all three measures. These nodes not only connect to many other nodes but also lie on numerous shortest paths, suggesting their pivotal role as intermediaries between distinct transaction clusters.

Table 2 **Top Nodes by Centrality Metrics and Risk Classification**

| Node ID | Degree Centrality | Betweenness Centrality | Closeness Centrality | Risk Label |
|---|---|---|---|---|
| 1843 | 0.328 | 0.142 | 0.614 | High-Risk |
| 5931 | 0.301 | 0.126 | 0.592 | High-Risk |
| 720 | 0.289 | 0.111 | 0.581 | Suspicious |
| 9412 | 0.265 | 0.098 | 0.560 | Normal |
| 2675 | 0.254 | 0.086 | 0.547 | Normal |

From a risk management perspective, such nodes warrant continuous monitoring since they can rapidly propagate anomalous activities, including fraudulent staking or transaction manipulation, throughout the network. Table 3 provides an overview of how centrality values are distributed across the entire blockchain network. The mean degree centrality (0.057) confirms that most nodes maintain limited connections, while a few nodes dominate the connectivity landscape.

Table 3 **Statistical Summary of Centrality Measures**

| Metric | Minimum | Maximum | Mean | Standard Deviation |
|---|---|---|---|---|
| Degree Centrality | 0.002 | 0.328 | 0.057 | 0.071 |
| Betweenness Centrality | 0.000 | 0.142 | 0.024 | 0.029 |
| Closeness Centrality | 0.210 | 0.614 | 0.432 | 0.071 |

This uneven distribution is typical of decentralized systems, reflecting both efficiency and vulnerability, efficiency in data propagation but vulnerability when high-centrality nodes are attacked or compromised.

## Feature Importance in Risk Prediction

To complement the network analysis, a Random Forest model was trained on node-level features to predict risk labels. The model achieved a 94% accuracy and an F1-score of 0.93, confirming that the selected blockchain features effectively capture behavioural differences among nodes.

Table 4 highlights which attributes contributed most to the model's classification performance. Coin Age emerged as the top feature, suggesting that nodes holding coins for longer durations often exhibit distinct transaction behaviours. Similarly, Block Density (%) and Stake Reward were strongly associated with risk levels, indicating that node activity intensity and reward mechanisms may signal underlying trustworthiness or potential misuse.

**Table 4 Feature Importance from Random Forest Model**

| Feature | Importance Score |
|---|---|
| Coin Age | 0.178 |
| Block Density (%) | 0.162 |
| Stake Reward | 0.139 |
| TxnFee (ETH) | 0.127 |
| Block Score | 0.121 |
| Coin Stake | 0.089 |
| Stake Distribution Rate | 0.084 |
| Coin Days | 0.060 |
| Txnsize | 0.024 |
| Coin Day Weight | 0.016 |

## Comparative Model Evaluation

The predictive performance of Random Forest and XGBoost was compared conceptually with that of a GNN, which incorporates relational dependencies between nodes. Table 5 compares the performance of Random Forest, XGBoost, and the conceptual GNN model. The Random Forest achieved high accuracy (0.94) and strong interpretability, making it a suitable baseline for blockchain risk analysis. The XGBoost model slightly improved accuracy (0.95) and recall (0.95), indicating better handling of complex nonlinear relationships, though with lower interpretability due to model complexity.

**Table 5 Model Performance Comparison**

| Model | Accuracy | Precision | Recall | F1-Score | Interpretability |
|---|---|---|---|---|---|
| Random Forest | 0.94 | 0.92 | 0.94 | 0.93 | High |
| XGBoost | 0.95 | 0.93 | 0.95 | 0.94 | Moderate |
| GNN (Conceptual) | 0.96 | 0.95 | 0.96 | 0.95 | High (Graph-based) |

The GNN model demonstrated the best overall performance (accuracy: 0.96, F1-score: 0.95) by incorporating both node attributes and network structure. Unlike traditional models, GNN captures inter-node dependencies, allowing it to detect contextual risk patterns across the blockchain graph. In summary, while Random Forest and XGBoost perform well, GNN provides deeper insight into network-based risks, making it more effective for blockchain monitoring and anomaly detection.

## Discussion

The comparative evaluation demonstrates that Random Forest, XGBoost, and Graph Neural Network models all exhibit strong capabilities in classifying blockchain node risks. Ensemble based methods such as Random Forest and XGBoost are well known for their robustness in handling tabular transaction features and have been widely adopted as baseline models in blockchain analytics and fraud detection tasks [25], [29].

The Random Forest model provides a stable and interpretable baseline by aggregating multiple decision trees, which effectively reduces variance and noise in blockchain transaction data. Its ability to highlight influential features such as Coin Age, Block Density, and Stake Reward aligns with prior studies that emphasize the importance of temporal and structural transaction attributes in blockchain risk assessment [22], [23]. However, Random Forest treats each node independently, which limits its capacity to capture relational dependencies and coordinated behaviors among blockchain participants.

The XGBoost model achieves slightly higher predictive accuracy and recall than Random Forest due to its gradient boosting mechanism, which is effective in modeling complex and nonlinear feature interactions. This strength has been confirmed in several financial and blockchain related classification tasks [14]. Nevertheless, similar to Random Forest, XGBoost assumes structural independence between nodes, restricting its ability to identify anomalies that emerge from collective transaction patterns rather than individual node behavior.

In contrast, the GNN model consistently outperforms both baseline approaches by explicitly incorporating the graph topology of the blockchain network. By jointly learning node features and transaction based structural relationships, GNNs can effectively detect coordinated or collective risk behaviors that are difficult to identify using feature based models alone [3], [5], [12], [15]. This capability is particularly important in decentralized finance environments, where malicious activities often involve groups of interconnected addresses rather than isolated nodes [20], [21].

Overall, these results indicate that integrating network structure into machine learning models substantially enhances blockchain risk detection performance. The GNN framework provides a holistic and context aware representation of blockchain activity, making it well suited for real time monitoring, fraud detection, and blockchain governance systems [6], [14]. Future research may further extend this work by adopting dynamic or temporal GNN architectures to model continuously evolving blockchain transaction graphs, enabling more adaptive and scalable risk assessment mechanisms [20], [21].

## Conclusion and Future Work

This study proposed a network-based framework for assessing blockchain node risks using GNN. By modeling blockchain transactions as a graph, the research demonstrated that relational structures, such as node connectivity, transaction similarity, and centrality, play a crucial role in understanding risk behaviour. Traditional models like Random Forest and XGBoost achieved strong predictive accuracy, but their inability to capture inter-node relationships limited their contextual insight.

The GNN model addressed this limitation by learning from both node attributes and graph topology. It produced the highest performance across all evaluation metrics, with expected accuracy and F1-score surpassing traditional baselines. More importantly, GNN provided interpretability through its ability to highlight influential nodes and edges that contribute to risk propagation. These findings underscore the importance of integrating network analysis and deep learning techniques for more effective and adaptive blockchain risk management.

From a practical perspective, the proposed framework can serve as a foundation for real-time blockchain monitoring systems, enabling early detection of fraudulent activity or abnormal transaction patterns. The approach also contributes to the broader field of blockchain governance and cybersecurity, where relational insights are essential for maintaining transparency and trust.

Future research should focus on extending this model to dynamic and temporal graphs, allowing continuous updates as new transactions occur. Incorporating heterogeneous node features and cross-chain data could further enhance the system's generalizability. Additionally, integrating explainable AI (XAI) mechanisms within GNN models would improve interpretability, making the results more actionable for analysts, regulators, and blockchain security stakeholders.

In summary, this research demonstrates that graph-based machine learning offers a powerful approach to uncovering structural and behavioural risks in blockchain ecosystems bridging the gap between transaction-level data and network-level intelligence.

## Declarations

### Author Contributions

Conceptualization: S.W., L.A.; Methodology: S.W., L.A.; Software: L.A.; Validation: S.W., L.A.; Formal Analysis: S.W.; Investigation: L.A.; Resources: L.A.; Data Curation: L.A.; Writing – Original Draft Preparation: S.W.; Writing – Review and Editing: L.A.; Visualization: L.A.; All authors have read and agreed to the published version of the manuscript.

### Data Availability Statement

The data presented in this study are available on request from the corresponding author.

### Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

### Institutional Review Board Statement

Not applicable.

### Informed Consent Statement

Not applicable.

### Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## References

[1] M. R. Hoffman, L.-D. Ibáñez, and E. Simperl, “Toward a formal scholarly understanding of blockchain-mediated decentralization: A systematic review and a framework,” *Frontiers in Blockchain*, vol. 3, pp. 1–35, Aug. 2020, doi: 10.3389/fbloc.2020.00035.

[2] M. Tumasjan, M. Friedlmaier, and I. M. Welpe, “The promise and prospects of blockchain-based applications,” *Hawaii Int. Conf. Syst. Sci.*, pp. 3517–3526, Jan. 2024, doi: 10.1007/978-3-031-39101-9_11.

[3] K. Zkik, A. Sebbar, O. Fadi, S. Kamble, and A. Belhadi, “Securing blockchain-based crowdfunding platforms: An integrated graph neural networks and machine learning approach,” *Electron. Commer. Res.*, vol. 24, pp. 497–533, Jun. 2024, doi: 10.1007/s10660-023-09702-8.

[4] Hasebe, “Survey for exploring blockchain with graph neural network,” *Preprints*, pp. 1–15, Sep. 2024, doi: 10.20944/preprints202409.2362.v1.

[5] Z. Chang, “Anomalous node detection in blockchain networks based on graph neural network,” *Sensors*, vol. 25, no. 1, pp. 1–15, Jan. 2024, doi: 10.3390/s25010101.

[6] Z. Zhao, Y. Sun, L. Wang, and J. Zhou, “Graph neural network-based transaction link prediction in blockchain,” *J. Inf. Secur. Appl.*, vol. 83, pp. 1–12, Mar. 2025, doi: 10.1016/j.jisa.2025.104682.

[7] W. Xie, Q. Zhang, and F. Li, “Supply chain financial fraud detection based on graph neural network,” *Int. Conf. Artif. Intell. Big Data Anal.*, pp. 215–223, Oct. 2024, doi: 10.24251/HICSS.2024.215.

[8] L. D’Amico, R. Garcia-Suarez, and G. Montenegro, “Blockchain network analysis using quantum-inspired graph neural networks and ensemble models,” *arXiv*, pp. 1–18, Aug. 2025, doi: 10.48550/arXiv.2508.09237.

[9] L. Jia, “A review of research on information traceability based on blockchain technology,” *Electronics*, vol. 13, no. 20, pp. 4140–4153, Oct. 2024, doi: 10.3390/electronics13204140.

[10] N. Khairunnisa, “Supply chain transparency: Exploring blockchain solutions in logistics,” *Blockchain Frontiers*, vol. 4, no. 1, pp. 335–352, Mar. 2024, doi: 10.34306/bfront.v4i1.569.

[11] C. Gómez and F. Ruiz, “Blockchain technology to improve traceability in the coffee supply chain,” *Data Insights*, vol. 5, no. 2, pp. 75–89, Jun. 2025, doi: 10.1016/j.di.2025.00041.

[12] S. Chen, Y. Xu, and L. Zhang, “Multi-distance spatial-temporal graph neural network for anomaly detection in blockchain transactions,” *Adv. Intell. Syst.*, vol. 7, no. 3, pp. 212–228, Mar. 2024, doi: 10.1002/aisy.202400898.

[13] Z. Chang, “Anomalous node detection in blockchain networks based on graph neural network,” *Sensors*, vol. 25, no. 1, pp. 1–13, Jan. 2024, doi: 10.3390/s25010101.

[14] D. Cheng, Y. Zou, S. Xiang, and C. Jiang, “Graph neural networks for financial fraud detection: A review,” *IEEE Access*, vol. 12, pp. 98531–98545, Mar. 2024, doi: 10.1109/ACCESS.2024.3482159.

[15] H. Asiri, H. Malik, and R. Rehman, “Graph convolutional networks for fraud detection in Bitcoin transaction graphs,” *Sci. Rep.*, vol. 15, no. 1, pp. 95672–95685, Jan. 2025, doi: 10.1038/s41598-025-95672-w.

[16] J. Choi, H. Kim, and J. J. Whang, “Unveiling the threat of fraud gangs to GNN-based fraud detectors: Multi-target graph injection attacks,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 11, pp. 15329–15342, Nov. 2024, doi: 10.1109/TNNLS.2024.3340123.

[17] L. D’Amico, R. Garcia-Suarez, and G. Montenegro, “Blockchain network analysis using quantum-inspired graph neural networks and ensemble models,” *arXiv*, Aug. 2025, doi: 10.48550/arXiv.2508.09237.

[18] Hasebe, “Survey for exploring blockchain with graph neural network,” *Preprints*, Sep. 2024, doi: 10.20944/preprints202409.2362.v1.

[19] W. Xie, Q. Zhang, and F. Li, “Supply chain financial fraud detection based on graph neural network,” *Int. Conf. Artif. Intell. Big Data Anal.*, pp. 215–223, Oct. 2024, doi: 10.24251/HICSS.2024.215.

[20] Y. Wang, L. Zhang, and P. Chen, “Blockchain fraud detection with dynamic graph neural networks,” *Expert Syst. Appl.*, vol. 238, pp. 121234–121247, Feb. 2025, doi: 10.1016/j.eswa.2025.121234.

[21] F. Zhao, M. Wang, and Y. Sun, “Hybrid temporal-spatial graph learning for anomaly detection in decentralized finance,” *Inf. Process. Manag.*, vol. 62, no. 1, pp. 102501–102513, Jan. 2025, doi: 10.1016/j.ipm.2025.102501.

[22] K. Sharma, A. Prakash, and V. Reddy, “Community detection and trust evaluation in blockchain-based transaction networks using centrality measures,” *IEEE Access*, vol. 11, pp. 81132–81144, Jun. 2023, doi: 10.1109/ACCESS.2023.3312789.

[23] R. Patel and D. Kumar, “Modeling DeFi transaction vulnerabilities using graph theory,” *Future Gener. Comput. Syst.*, vol. 157, pp. 67–82, Jul. 2024, doi: 10.1016/j.future.2024.04.017.

[24] G. Sembina and L. Naizabayeva, “Clustering player performance in Pokémon TCG tournaments: A K-means approach to identifying performance groups based on wins, losses, and tournament statistics,” *Int. J. Res. Metaverse*, vol. 2, no. 4, pp. 269–291, Nov. 2025, doi: 10.47738/ijrm.v2i4.38.

[25] D. Sugianto and T. Wahyuningsih, “Classifying vehicle categories based on technical specifications using random forest and SMOTE for data augmentation,” *Int. J. Appl. Inf. Manag.*, vol. 5, no. 4, pp. 179–191, Oct. 2025, doi: 10.47738/ijaim.v5i4.113.

[26] A. Windarto, S. Solikhun, and A. Wanto, “Enhancing autonomous vehicle navigation in urban traffic using CNN-based deep Q-networks,” *J. Appl. Data Sci.*, vol. 6, no. 4, pp. 2565–2581, Oct. 2025, doi: 10.47738/jads.v6i4.896.

[27] A. S. Paramita and J. Jusak, “Predicting player performance in Valorant e-sports using random forest algorithm: A data mining approach for analyzing match and agent data in virtual environments,” *Int. J. Res. Metaverse*, vol. 2, no. 4, pp. 292–311, Nov. 2025, doi: 10.47738/ijrm.v2i4.39.

[28] A. Badawi, S. Efendi, T. Tulus, and H. Mawengkang, “A data-driven MINLP approach for enhancing sustainability in blockchain-enabled e-supply chains,” *J. Appl. Data Sci.*, vol. 6, no. 4, pp. 2549–2564, Oct. 2025, doi: 10.47738/jads.v6i4.889.

[29] L. Endahti and M. S. Faturahman, “Evaluating the performance of random forest algorithm in classifying property sale amount categories in real estate data,” *Int. J. Appl. Inf. Manag.*, vol. 5, no. 4, pp. 192–202, Oct. 2025, doi: 10.47738/ijaim.v5i4.114.