

A Performance Comparison of Graph Neural Networks and Standard Machine Learning Methods for Illegal Bitcoin Transaction Detection Based on the Elliptic Dataset

Ruiqi He*

School of Computer Science, Guangdong Polytechnic Normal University, Guangzhou, 510665, China

Abstract. Bitcoin allows users to complete transactions anonymously and operates without centralized management. Such inherent characteristics provide conditions for various illegal activities. Global financial safety is threatened by these unlawful behaviors in virtual currency markets. Traditional machine learning models rely solely on the inherent attributes of single transactions for analysis. The hidden topological links existing in transaction networks cannot be captured by such methods. Such drawbacks prevent traditional detection tools from producing qualified identification outcomes. This study takes the Elliptic Bitcoin dataset as the basic experimental material. Three mainstream graph neural network models are adopted in the experiment. Random Forest and XGBoost are also included for parallel testing. The dataset is divided by time step to prevent temporal data leakage, and class weights are uniformly applied to all models to address extreme label imbalance. The detection capacity of all selected algorithms is compared in a unified experimental environment. Higher detection accuracy for Bitcoin illegal transactions is shown by graph neural network models. Among all tested models, Graph Isomorphism Network (GIN) achieves the highest Recall and F1-score on the illicit class, demonstrating superior capacity to learn structural transaction patterns.

1 Introduction

Blockchain technology has expanded quickly in recent times. Bitcoin and other digital currencies have gained huge popularity, and they have changed how global financial activities operate [1]. Digital currency platforms hide user identities and run without central control. They can also be used for transactions across different regions. These natural traits create convenient conditions for illegal acts. Money laundering, fraud, and terrorist funding all take place frequently on such platforms [2].

Common machine learning models like Random Forest are often used for illegal transaction detection. These models only collect feature data from individual transaction nodes. They cannot identify hidden connection structures inside cryptocurrency transaction networks [3]. Such shortcomings make these tools fail to find covert illegal deals and group

* Corresponding author: 2024036643031@stu.gpnu.edu.cn

trading behaviors accurately. The models collect feature information from surrounding nodes and update internal parameters continuously [4]. This study uses the well-known Elliptic Bitcoin Dataset to carry out controlled experiments. Three mainstream Graph Neural Network (GNN) models, including GIN, Graph Attention Network (GAT), and Graph Convolutional Network (GCN), are chosen for performance testing and comparison [5].

The Elliptic Bitcoin Dataset builds network topology according to real Bitcoin trading logic. Each node in the network stands for a single Bitcoin transaction. Edges are set to record direct capital transfer behaviors between trading users [6]. This study sets binary classification as the main research task. It aims to tell legal transactions apart from illegal ones within the dataset. The transaction network contains a large number of nodes and presents sparse connection features. The use of these structural links can raise the precision and recall level of illegal transaction identification. The sample labels in this dataset are distributed unevenly. Legal transaction samples occupy the main proportion, while illegal samples are much fewer in number. Model training will lean toward the majority samples under such an unbalanced distribution. The recognition effect of illegal transactions will drop as a result, and the whole process of model training and result evaluation becomes more difficult.

2 Related work

Early researchers have focused on developing methods to identify illegal transaction activities on the Bitcoin network. They mostly adopted manually designed feature sets and standard supervised machine learning classification algorithms [7]. Such methods only capture the individual characteristics of single nodes, ignoring the fund flow logic and topological connection relationships in the transaction network, resulting in poor adaptability to complex and evolving illicit transaction patterns. With the continuous development of graph data analysis and deep learning technology, recent research has gradually shifted to GNN-based illicit transaction detection methods, which break through the limitations of traditional methods by deeply mining the topological connection patterns of transaction networks [8]. Relevant experiments have verified that GNN models can effectively aggregate neighborhood feature information of transaction nodes and mine the structural characteristics of transaction graphs, achieving obviously better detection results than traditional machine learning models.

A large number of studies have confirmed that GNNs are far superior to traditional machine learning methods in processing graph-structured transaction data and identifying illicit transaction patterns [9]. The unique message-passing mechanism of GNNs makes them naturally suitable for processing transaction graph data and can effectively capture the relational features between transaction nodes that are difficult to extract by traditional methods [10, 11].

Although existing studies have fully proved the effectiveness and superiority of GNNs in Bitcoin illicit transaction detection, most of the current research focuses on the research and application of a single GNN model, and there is a lack of rigorous and systematic comparative research on the detection performance of different GNN architectures based on the same dataset and experimental settings. Different GNN models have distinct core feature aggregation mechanisms: GCN adopts spectral-based neighborhood feature aggregation, GAT introduces an attention mechanism to realize adaptive weighting of neighboring nodes, and GIN is based on the Weisfeiler-Lehman graph isomorphism test to capture fine-grained structural features of transaction graphs. Few existing works carry out full side-by-side tests on mainstream graph neural network structures in real Bitcoin data environments.

3 Methodology

3.1 Data preprocessing

This experiment adopts the Elliptic Dataset collected from Kaggle. The research selects basic experimental indicators such as transaction volume, time step information, node centrality degree, adjacent connection relationships, and 165 kinds of multidimensional transaction attributes. Basic data processing steps contain data cleaning, missing value filling, reasonable category screening, unified feature normalization, and complete graph network building. The paper removed all samples marked with unknown tags in the screening process. The whole data set is divided according to time sequence to build training subsets and testing subsets. The original unbalanced sample ratio of the dataset is kept completely unchanged to guarantee real and objective experimental outcomes. All tested algorithms contain GCN, GAT, GIN, Random Forest, and XGBoost. The same class weight parameter is set for every model to ensure fair comparison. Records from the first time step to the 34th time step are put into the training set, and data from the 35th time step to the 49th time step are used for model testing.

3.2 Model architectures

This study implements and compares three classical graph neural network models, namely GCN, GAT, and GIN, under completely consistent experimental environments, parameter frameworks, and training strategies. All three models take the preprocessed graph structure and node features as input and output the classification probability of licit or illicit for each transaction node.

GCN conducts feature aggregation based on spectral graph convolution. It normalizes the adjacency matrix and aggregates the feature information of neighbor nodes in a fixed and equal-weight manner, which can effectively capture the local topological structure of the transaction graph and realize the preliminary learning of fund flow rules.

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

The node feature matrix at layer l is denoted as $H^{(l)}$. The adjacency matrix with added self-loops is represented by $\tilde{A} = A + I$, while $\tilde{D}$ corresponds to the diagonal degree matrix of $\tilde{A}$. At layer l , $W^{(l)}$ is the trainable weight matrix, and σ refers to the activation function, such as Rectified Linear Unit (ReLU).

GAT introduces an adaptive attention mechanism on the basis of GCN. It calculates the attention coefficient between nodes through a learnable attention vector, automatically assigns different weights to different neighbor nodes, and enhances the perception of key transaction paths and suspicious capital flows, so as to improve the recognition accuracy of concealed illegal transactions.

$$h_i^{(l+1)} = \sigma \left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(l)} W^{(l)} h_j^{(l)} \right) \quad (2)$$

In the GAT layer update rule, $h_i^{(l)}$ denotes the feature vector of node i at layer l . $\mathcal{N}(i)$ represents the set of neighbors of node i , while $\alpha_{ij}^{(l)}$ is the learned attention coefficient between node i and its neighbor node j . $W^{(l)}$ is the trainable weight matrix used to linearly transform node features at layer l , and σ denotes the non-linear activation function applied after aggregation.

$$\alpha_{ij}^{(l)} = \frac{\exp(\text{LeakyReLU}(a^T [W^{(l)} h_i^{(l)} \| W^{(l)} h_j^{(l)}]))}{\sum_{k \in \mathcal{N}(i)} \exp(\text{LeakyReLU}(a^T [W^{(l)} h_i^{(l)} \| W^{(l)} h_k^{(l)}]))} \quad (3)$$

In the attention coefficient formula, α is the trainable attention parameter vector, and $||$ denotes the concatenation operation for combining node features.

GIN is constructed based on the Weisfeiler-Lehman graph isomorphism test, which has the strongest expression ability among the three models. It uses a Multilayer Perceptron (MLP) for feature transformation and iterative aggregation, which can capture fine-grained subgraph structure features that are difficult to distinguish by GCN and GAT.

$$h_i^{(l+1)} = MLP^{(l)} \left((1 + \epsilon) \cdot h_i^{(l)} + \sum_{j \in N(i)} h_j^{(l)} \right) \quad (4)$$

In the GIN update rule, $h_i^{(l)}$ is the feature vector of node i at layer l , and $N(i)$ is the set of its neighbors. The parameter ϵ is a learnable or fixed scalar that adjusts the contribution of the node's own features, while $MLP^{(l)}$ is the multi-layer perceptron at layer l that transforms the aggregated feature into $h_i^{(l+1)}$.

By fully mining the unique topological characteristics of illegal transaction clusters, GIN can better identify the structural differences between legal and illegal transactions and show stronger stability and accuracy in dealing with complex and diverse illegal transaction patterns.

3.3 Evaluation metrics and analysis pipeline

In order to comprehensively, objectively, and fairly evaluate the performance of each model in illegal transaction detection tasks, this study constructs a multi-dimensional evaluation system including quantitative indicators, efficiency indicators, and qualitative analysis.

In terms of quantitative evaluation, core indicators closely related to anti-money laundering scenarios are selected, including Accuracy, Precision, Recall, F1-score, and Area Under the Receiver Operating Characteristic Curve (AUC-ROC).

Accuracy reflects the overall proportion of correctly classified samples among all samples, and is used to measure the overall prediction correctness of the model. The formula is:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

Where True Positive (TP) is the number of illegal transactions correctly identified, True Negative (TN) is the number of normal transactions correctly identified, False Positive (FP) is the number of normal transactions incorrectly identified as illegal, and False Negative (FN) is the number of illegal transactions incorrectly identified as normal.

Precision measures the proportion of truly illegal transactions among all transactions predicted as illegal by the model, which reflects the "purity" of the model's suspicious transaction screening results. The formula is:

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

High precision values in the experiment show that most transactions marked as suspicious by the model belong to real illegal cases. Fewer irrelevant tasks will be arranged for manual inspection with the support of such a model's performance. Recall is calculated based on the number of real illegal transactions. It shows how many actual illegal records are successfully found by the model. The paper can directly determine whether the model can fully locate all hidden illegal behaviors through this indicator. The formula is:

$$Recall = \frac{TP}{TP+FN} \quad (7)$$

Recall plays an essential role in all anti-money laundering work. The overall detection ability of the model for high-risk illegal transactions is decided by this core evaluation index. The paper uses the F1-score to combine precision and recall in experimental tests. This index takes both evaluation dimensions into full consideration. It prevents unfair judgment results when experiments only rely on a single assessment standard to judge model performance. The formula is:

$$F1-score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (8)$$

AUC-ROC stands for the Area Under the Receiver Operating Characteristic curve. It calculates the total area below the ROC curve. The paper draws the curve by matching true positive rates with false positive rates under different classification thresholds. Models with higher AUC values can better separate normal transactions from illegal ones. Accuracy does not serve as a major judgment standard in the whole experiment. It is only treated as extra reference data. Recall and F1-score are chosen as core evaluation standards. These two indices prevent unfair assessment results brought by imbalanced sample sizes. The paper records training time and inference speed in detail. The t-distributed Stochastic Neighbor Embedding (t-SNE) visualization method is applied in qualitative analysis. It clearly shows the distance between the embedding vectors of legal and illegal transactions. The paper also studies prediction data distributions and confusion matrices. These materials reflect the actual prediction confidence and common error types of each algorithm.

4 Results and analysis

4.1 Exploratory data analysis results

The paper performs exploratory data analysis before all model training steps. The work uncovers the basic structural features of the Elliptic Bitcoin transaction network.

The paper observes the overall network data in Fig. 1. The whole network presents standard power-law degree distribution features. Most nodes in the network have low connection values. Only a small number of nodes carry large amounts of connection links and serve as key network hubs.

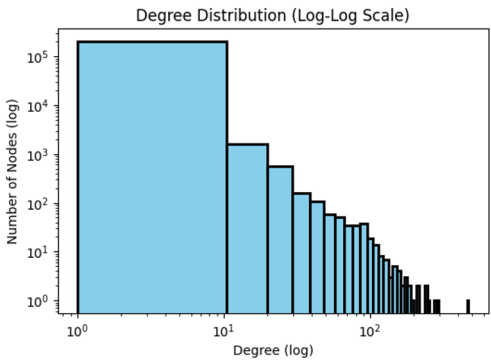


Fig. 1. Degree distribution of the Bitcoin transaction network (Photo credit: Original).

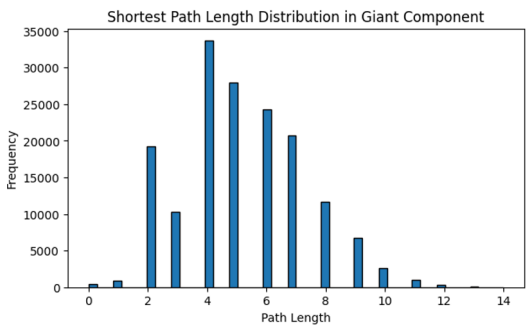


Fig. 2. Distribution of shortest path lengths in the transaction network (Photo credit: Original).

The distribution of shortest path lengths can be seen in Fig. 2. It shows that the Bitcoin transaction network has small-world properties. Most connection paths in the network range from 4 to 6 steps. It also lets illegal transaction behaviors spread quickly and stay hidden from regular detection.

The paper carries out centrality tests and displays the results in Fig. 3. A small number of core nodes are found to control most fund movements in the transaction network. The visual results further prove that nodes with high centrality values hold the whole network connected. These key nodes link regular transaction data and illicit transaction data together in the whole network.

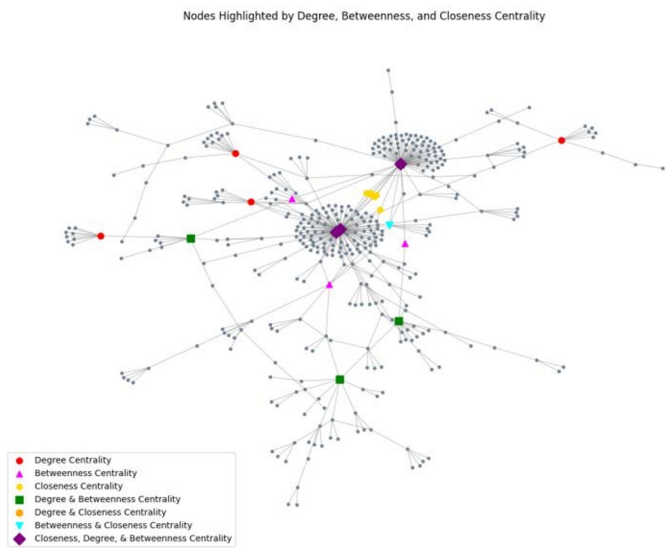


Fig. 3. Transaction nodes highlighted by degree centrality, betweenness centrality, and closeness centrality (Photo credit: Original).

Community detection results are presented in Fig. 4. Obvious structural gaps exist between groups of different transaction types. Groups made up of legal transactions grow to a large scale with close internal connections and a centralized layout. Groups formed by illegal transactions stay small in size with loose connections and no fixed central structure. GNN models can learn useful reference rules from these different structural features.

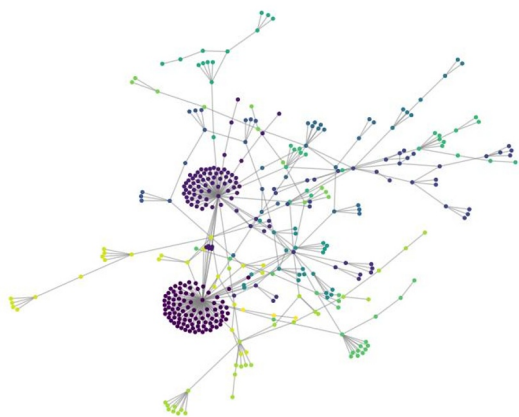


Fig. 4. Community structure comparison between licit and illicit transaction clusters (Photo credit: Original).

4.2 Quantitative performance comparison

Fig. 5 lists the test results of all involved algorithms. GIN records the steadiest comprehensive level across all evaluation metrics. The model obtains a recall of 0.86, an F1-score of 0.90, and a precision of 0.93. Acceptable accuracy can be maintained by both GCN and GAT, but neither model achieves a comparable recall performance.

XGBoost and Random Forest are typical tree-based machine learning models. Stable accuracy and precision results are produced by the two models in practical experimental tests. Poor performance is shown in their recall indicators during the whole detection process. These models depend on manually designed feature sets for analysis. They ignore the connection topology between transaction nodes. The design makes them unable to find sparse and hidden money laundering behaviors. Graph neural network models learn internal relations of nodes directly from graph data. GIN performs best among these methods. This model is able to find complex hidden relational paths within sets of illegal transaction records.

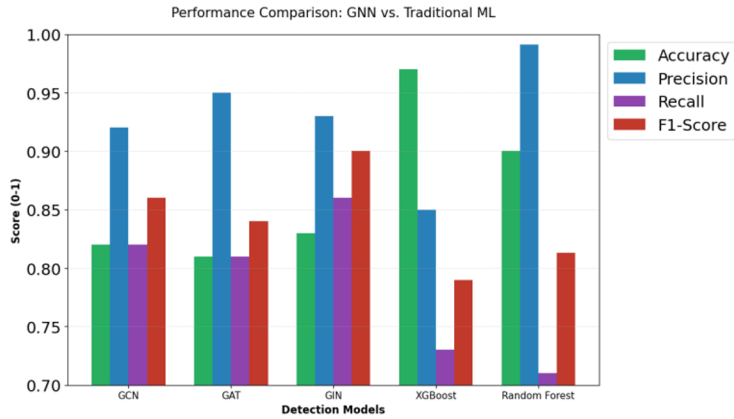


Fig. 5. Performance comparison of GNN and Traditional ML (Photo credit: Original).

The box plot in Fig. 6 displays predicted probability values for each model. Model outputs for legal transactions gather close to 0.0 in the GIN results. The model maintains stable confidence in its judgment. Output ranges become more scattered for GCN and GAT on the same transaction type. Most prediction values of GIN stay above 0.8 when processing illegal transaction samples. Obvious gaps form between the two transaction categories in the distribution. GIN can learn diverse feature representations during the training process. Compared with GCN and GAT, the model has a stronger ability to tell different transaction categories apart in practical detection tasks.

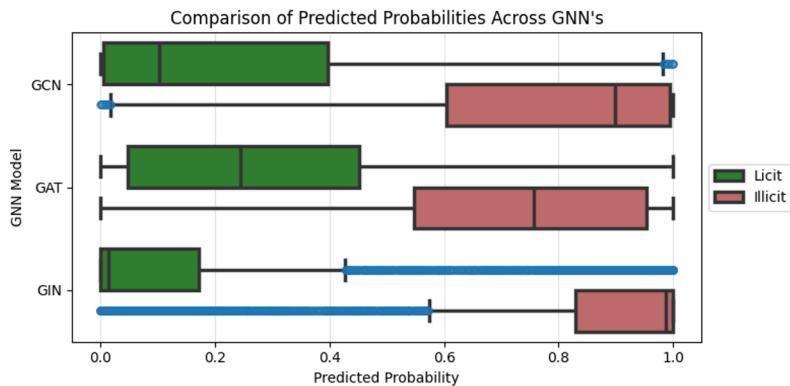


Fig. 6. Box plots of predicted probabilities for licit and illicit transactions (Photo credit: Original).

4.3 Qualitative analysis via t-SNE visualization

Node embedding results from the GIN model are presented in the t-SNE visualization of Fig. 7. Clear separation is observed between data points of legal transaction samples and illegal transaction samples in the visual distribution. Most areas of the embedding space are occupied by legal transaction nodes, and these nodes form large and compact distributed clusters. Illegal transaction nodes are gathered in small independent regions, and researchers can easily separate these nodes from other surrounding sample points. The clear spatial separation in the visual chart is consistent with the objective test scores of the GIN model in practical experiments.

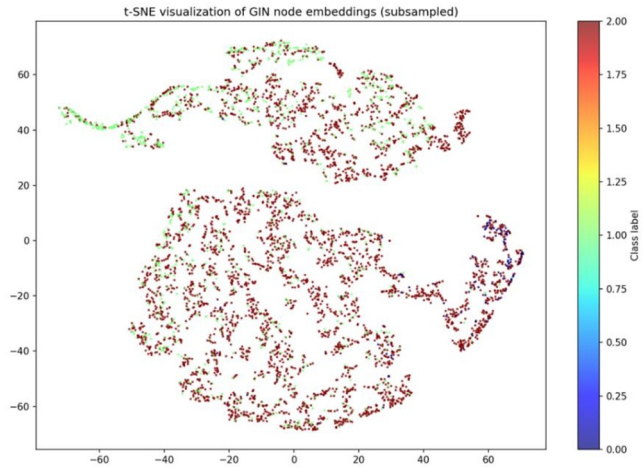


Fig. 7. t-SNE visualization of node embeddings learned by GIN (Photo credit: Original).

4.4 Model correlation and comprehensive analysis

Relevant correlation data are shown in Fig. 8. The changing trends of prediction probability remain close across the three graph neural network models. Every correlation coefficient calculated in the experiment remains above 0.776. The three models still show different performance when distinguishing various transaction types. GIN achieves better classification outcomes than GAT, while GAT obtains better test results than GCN.

Differences exist in the way each model extracts and expresses feature information. Fine local subgraph structures within transaction networks can be captured by GIN. The attention mechanism is adopted by GAT to highlight key connection relationships between nodes. GCN adopts a fixed weighting strategy and assigns equal importance to all neighbor node data during feature aggregation.

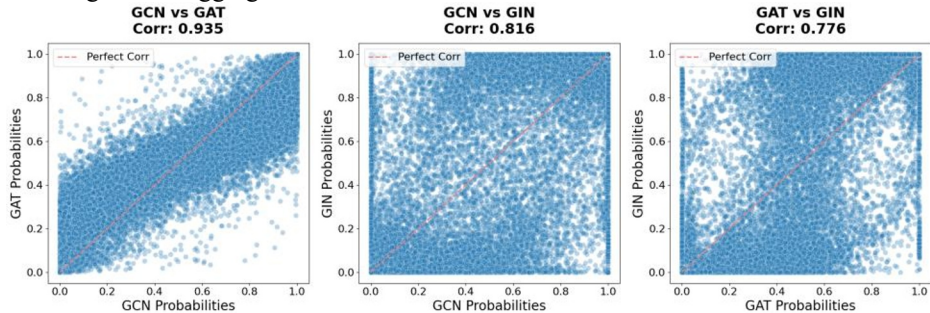


Fig. 8. Correlation scatter plots of prediction probabilities (Photo credit: Original).

GIN earns the top overall level across all evaluation metrics. It keeps steady precision values and forms clear feature divisions between sample groups. GAT maintains a good balance between detection performance and running efficiency. GCN can be used as a basic reference model for related experiments.

5 Conclusion

This work selects the Elliptic dataset to study Bitcoin illegal transaction detection. Three common graph neural network models, including GCN, GAT, and GIN, are compared alongside two classic machine learning methods, XGBoost and Random Forest. The experiment adopts the standard time-step division scheme with steps 1 to 34 used for training and steps 35 to 49 reserved for testing, preventing temporal leakage. Unified class weight parameters are configured across all five models to maintain experimental fairness under extreme label imbalance.

Graph neural networks can capture hidden topological features inside transaction networks from experimental outcomes. These models obtain better scores on core evaluation metrics than traditional machine learning methods. GIN shows the best overall detection ability and generates obvious feature differences between different transaction types. GAT and GCN also show usable application value with their own strengths in prediction accuracy and running speed. t-SNE visualization further confirms that GIN produces the clearest separation between licit and illicit node embeddings among all GNN architectures evaluated.

The research proves the practical value of graph neural networks in blockchain anti-money laundering scenarios. Future research will aim to create streamlined GNN architectures to speed up model inference. Self-supervised learning techniques will be integrated into the training process to reduce reliance on manually labeled samples. More experimental trials will be arranged to verify model stability and detection capacity. These tests will be set under heavier label imbalance and more complex transaction environments.

References

1. A. Venčkauskas, Š. Grigaliūnas, L. Pocius, R. Brūzgienė, A. Romanovs, Machine learning in money laundering detection over blockchain technology. *IEEE Access* **13**, 7555–7573 (2024)
2. Z. Chen, L.D. Van Khoa, E.N. Teoh, A. Nazir, E.K. Karuppiyah, K.S. Lam, Machine learning techniques for anti-money laundering (AML) solutions in suspicious transaction detection: A review. *Knowl. Inf. Syst.* **57**(2), 245–285 (2018)
3. I. Alarab, S. Prakoonwit, M.I. Nacer, Comparative analysis using supervised learning methods for anti-money laundering in bitcoin, in *Proceedings of the 2020 5th International Conference on Machine Learning Technologies* (2020), pp. 11–17
4. G. Corso, H. Stark, S. Jegelka, T. Jaakkola, R. Barzilay, Graph neural networks. *Nat. Rev. Methods Primers* **4**(1), 17 (2024)
5. N. Nimmagadda, M. Madijagan, Graph neural networks for illicit transaction detection in the Elliptic Bitcoin dataset, in *Proceedings of the 2025 5th International Conference on Evolutionary Computing and Mobile Sustainable Networks (ICECMSN)* (IEEE, 2025), pp. 1121–1127
6. S. Marasi, S. Ferretti, Anti-money laundering in cryptocurrencies through graph neural networks: A comparative study, in *Proceedings of the 2024 IEEE 21st Consumer Communications & Networking Conference (CCNC)* (IEEE, 2024), pp. 272–277

7. M. Weber, G. Domeniconi, J. Chen, D.K.I. Weidele, C. Bellei, T. Robinson, C.E. Leiserson, Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics. *arXiv:1908.02591* (2019)
8. A. Asiri, K. Somasundaram, Graph convolution network for fraud detection in bitcoin transactions. *Sci. Rep.* **15**(1), 11076 (2025)
9. A. Ancelotti, C. Liason, Review of blockchain application with graph neural networks, graph convolutional networks and convolutional neural networks. *arXiv:2410.00875* (2024)
10. Z. Zuo, Y. Jiang, R. Liang, J. Xu, H. Jiang, S. Zhang et al., A Bayesian-optimized XGBoost approach for money laundering risk prediction in financial transactions. *Information* **17**(4), 324 (2026)
11. Z. Xie, X. Huang, A credit card fraud detection method based on Mahalanobis distance hybrid sampling and random forest algorithm. *IEEE Access* **12**, 162788–162798 (2024)