ECTI Transactions on Computer and Information Technology
Journal homepage: https://ph01.tci-thaijo.org/index.php/ecticit/
Published by the ECTI Association, Thailand, ISSN: 2286-9131

Detecting Manipulation in the NFT Market Using Graph-based Deep Learning

Ade Indriawan[1] and Nur Aini Rakhmawati[2]

ABSTRACT

The rise of non-fungible tokens (NFTs) has increased the risk of fraud and market manipulation. This study introduces a method for detecting wash trading in the NFT marketplace using Graph Neural Networks (GNNs) applied to Ethereum blockchain transaction data. We constructed a heterogeneous graph, used Depth-First Search for labelling, and extracted graph features, including PageRank and degree centrality. We evaluate various classification models: Multilayer Perceptron (MLP), Graph Convolutional Neural Network (GCN), and Heterogeneous Graph Convolutional Neural Network (HeteroGCN). The results show that GNN models, particularly the feature-enhanced HeteroGCN, exhibit superior performance compared to featureless models and traditional tabular baselines. The key contribution of this study is that PageRank and Degree Centrality features significantly improve the accuracy of identifying transactions involved in market manipulation.

Article information:

Keywords: Fraud Detection, Feature Extraction, Graph Neural Network, Market Manipulation, NFT, Wash Trading

Article history:
Received: November 16, 2024
Revised: October 2, 2025
Accepted: October 23, 2025
Published: October 31, 2025
(Online)

DOI: 10.37936/ecti-cit.2025194.259882

1. INTRODUCTION

The use of non-fungible tokens (NFTs) as blockchain-based proof of ownership for digital assets is growing in popularity. NFTs allow creative content providers to generate money outside established commercial structures. Meanwhile, the public views NFTs as an alternative investment, expecting their value to rise in the future.

Despite their growing popularity, concerns persist about the high levels of speculation and fraud in NFT trading and investing [1]. The implementation of blockchain technology, such as cryptocurrencies and NFTs, can facilitate money laundering and terrorist financing [2] and reduce the trafficking of illegal drugs [3]. Trading in the Bitcoin and NFT markets also carries the risk of becoming involved in pump-and-dump operations [4], [5] and Ponzi schemes [6], [7].

Following a series of viral events, such as Ghozali Everyday, in which a young man sold hundreds of collections of daily selfie photographs captured over the last five years as NFTs, cryptocurrency and NFT fever spread throughout Indonesia [8]. People have since begun to battle for their digital assets across several NFT-specific marketplaces, despite a still-low level of public understanding of NFTs [9].

Some public figures profit from token and coin sales due to a lack of public knowledge about NFTs and crypto [10]. Because of their social media celebrity status, they were able to swiftly sell tokens to the public despite lacking any strategy that the public could actually use. Celebrity token sales typically rely on the fear of missing out (FOMO) hype from fans, using pump-and-dump tactics. As a result, several of these coins saw their prices plummet compared to when the tokens were issued [11]. The declining values of these tokens have resulted in significant losses for those who have already purchased them.

This study aims to provide an alternative technique for detecting NFT market manipulation using wash-trading schemes. Wash trading is a manipulation strategy in which investment products are bought and sold without any actual transfer of ownership [12]. This method is frequently used in pump-and-dump schemes to artificially inflate the price of an asset. As no unlawful acts are conducted, such as disseminating financial hoaxes or mass raising to target specific effects, as in the GameStop short squeeze [13], manipulation using wash trading techniques is considerably more challenging to identify. Traders execute wash trades through legitimate trading activ-

[1,2] The authors are with the Department of Information Systems, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia, Email: hey@adein.dev and nur.aini@is.its.ac.id
[2] Corresponding author: nur.aini@is.its.ac.id

ities. By designing a specific sequence of buying and selling actions, manipulators can perform transactions as desired [14]. Before implementing the wash-trading scheme, the manipulators agreed to form a network to facilitate transactions. Therefore, identifying such colluding networks is the most critical step in detecting wash trading activities [15].

Another difficulty is that real-world data, such as the datasets used in this study, may contain a high degree of dimensions. On the one hand, high-dimensional data can reveal a wealth of information. However, the high dimensionality of the data introduces significant redundancy and noise [16]. Dimensionality reduction can improve classification accuracy, minimise redundancy and noise, and simplify the learning methods. Some dimensionality reduction techniques used include feature extraction and selection methods [17]. In practice, identifying and acquiring features is not easy.

Furthermore, the dataset used in this study consisted of historical transaction data. The most critical data elements in the dataset are seller accounts, buyer accounts, transaction values, and the NFT objects transacted. The data lack any attributes that would help identify market manipulation, as they are too general and nonspecific. The dataset cannot yet be used as input for detecting market manipulation because the required fraud-detection attributes are not available. Consequently, an additional feature engineering process is needed to perform transaction categorisation.

The feature engineering process requires special measures to isolate relevant features and separate them from the irrelevant features. In the graph-formatted NFT dataset, selecting appropriate features can be achieved by applying graph algorithms, such as community detection and centrality measures. Therefore, in this study, we propose a method for detecting manipulation in the NFT market using a GNN with labelled data stored in the form of graphs.

Data labelling, performed independently, requires verification against the processed data labels. The features extracted from the labelled dataset are then grouped into node-, edge-, and graph-based features. Thus, the autoencoder helps isolate and ignore irrelevant features. In the last step, a Graph Neural Network is used to classify accounts and addresses as fraudulent or non-fraudulent based on their transaction history.

This study makes the following three primary contributions to the field of NFT market manipulation detection.

- Validate the HeteroGCN model's superior performance for classifying manipulation within graphically represented NFT data.
- Establish an objective DFS closed-loop detection method to create verifiable ground truth for fraudulent transactions.
- Identify key graph features that are essential for accurately predicting wash trading activity.

2. LITERATURE REVIEW

2.1 Market manipulation

Market manipulation is identical to the price manipulation of exchanged items [18]. This manipulation is not a new problem; it has existed for decades and will continue to do so. A purposeful attempt to steer the market price of an asset from its fair price is called price manipulation. Market manipulation is usually carried out to profit financially.

Various manipulation methods may affect an asset's market price [19].

1. Action-based manipulation. Manipulators misuse assets to manipulate their values.
2. Information-based manipulation: This method involves spreading incorrect or misleading information to move prices in the intended direction.
3. Trade-based manipulation. Individuals buy or sell a particular number of assets with the hope that prices will move in the desired direction because of their understanding of asymmetric information, the trading process, or inventory costs.

All three manipulation methods occurred in the commodity markets. Historically, the most significant type of manipulation has been market power manipulation (MPM), also known as action-based manipulation.

In a conventional financial system, a single centralised authority, such as banks and stock exchanges, can supervise all transactions. Such supervision is proper for monitoring, for example, whether a customer opens a new account or whether a large number of transactions occur [12]. However, this kind of supervision does not exist in a decentralised financial system, so everyone is free to open a new account (or wallet) and make large numbers of transactions without third-party oversight. The vulnerability of the decentralised financial system to misuse for illegal activities is an issue. However, anyone can easily view all transaction details stored on the blockchain network due to the technology's transparency. An increasing number of people are conducting transactions involving cryptocurrencies or NFTs on blockchain networks, making the detection of illegal activities difficult and time-consuming.

Several studies have examined various modes of fraud and scams in cryptocurrency and NFT trading markets. These modes include pump-and-dump [4], [20], phishing [21], wash trading [22], Ponzi schemes [23], and money laundering [24].

Wash trading is the market manipulation scheme that the study focuses on. According to the US Securities and Exchange Commission (SEC), wash trading is a securities transaction that does not involve a change in the ownership benefits of the securities in question [12]. Individuals involved in this scheme conduct transactions without any actual asset transfers. The purpose of the action is to create an illusion that traders are trading assets so that ordinary investors are interested in buying them. This scheme is often part of a pump-and-dump scheme.

2.2 Fraud/anomaly detection

Anomaly detection is the process of identifying patterns in data that do not conform to expected behaviour [25]. Anomaly detection aims to identify unexpected and rare events. Since these rare events or entities deviate from standard behaviour and patterns, they stand out as anomalies. In data, anomalies often appear as deviations from the mean (standard deviations), outliers, noise, novelties, or exceptions. Anomaly detection is based on two assumptions [26]: first, anomalies appear very rarely in the data; second, the characteristics of the data with anomalies are significantly different from those of the other data.

Anomaly detection plays a vital role in several fields, including market manipulation [27]. The growing complexity of the problem, with unique challenges, requires new, more advanced approaches to anomaly detection.

Several studies identifying market manipulation have presented strategies to detect such actions, particularly wash trading, using various methods. [14] employed graph analysis in conjunction with dynamic programming to identify possible wash trading activities. Graph analysis maps the structure of the traders involved. A new method identifies wash trades in financial instruments traded on NASDAQ and the London Stock Exchange by detecting suspicious matching orders and collusive behaviour among traders who submitted them. Both steps use a dynamic programming approach. A separate study found that wash trading activities have various transaction topologies, including ring, star, tree, and mesh topologies [15].

Furthermore, several studies have corroborated that transaction patterns in wash trading activities resemble ring topologies or closed loops [28]. Closed loops in manipulation may be closed-loop token trades or closed-loop value trades. Using the Depth-First Search (DFS) algorithm on the transaction graph, graph analysis can identify wash trade activities that generate closed-loop token trades [29].

In recent years, machine/deep learning has emerged as a promising method for detecting anomalies, particularly for market manipulation [30]. Machine learning algorithms were commonly used in these studies [31], [32] to create transaction-classification models. In some studies, data graphs have been combined with deep learning to analyse transactions and detect anomalies [24], [33]–[37].

2.3 Graph Neural Network

Connectivity is the most prevalent feature of current networks and systems worldwide. Networks of any complexity — from molecular interactions to social networks, communication systems to power grids, and shopping experiences to supply chains — are not random, indicating that connections are neither equally distributed nor static. Simple statistical analysis alone cannot adequately describe the behaviour of interconnected systems, let alone make accurate predictions [38]. A network or graph is a data structure representing a collection of objects (nodes) and their relationships (edges) [39]. Data modelling using graphs is an ideal technique for analysing data stored in blockchains, given their connected structure.

A Graph Neural Network is a neural network technique that processes data in graph form [40]. Graphs are essential data structures because they can represent real-world problems for easy analysis. Examples of graph implementations help solve real-world problems, such as those in social networks, geographic maps, and web page links.

A Graph Neural Network is a method that has recently been implemented in several studies [41]. This method is based on deep learning and operates on graph-structured data. Owing to its ability to process graphs, a Graph Neural Network is widely used in research on anomalies and fraud detection using network data [21], [37].

Several studies have used Graph Neural Networks to detect scams and fraud in blockchain-based digital asset trading. However, these studies still focus on detecting scams or fraud in transactions in the cryptocurrency market [5], [36], [42]. Therefore, this study aims to fill this gap.

3. DATA AND METHOD

The dataset used in this study comprises historical data on NFT buying and selling transactions from April 1 to September 25, 2021. In total, the dataset contains 15 tables, and the transfer Table, which serves as the input to the classification model, contains 4,514,729 transactions. The dataset comprises unlabeled data retrieved directly from the Ethereum blockchain network. Therefore, data labelling was one of the main processes performed in this study. The process of labelling data involves implementing a graph algorithm, namely Depth-First Search, to detect a cycle or closed loop in a series of buying and selling transactions for an NFT object.

Before the data labelling process, the historical data must be grouped by each NFT object's transactions, which are then represented as graphs generated via graph data modelling. The flowchart of the study is shown in Figure 1.

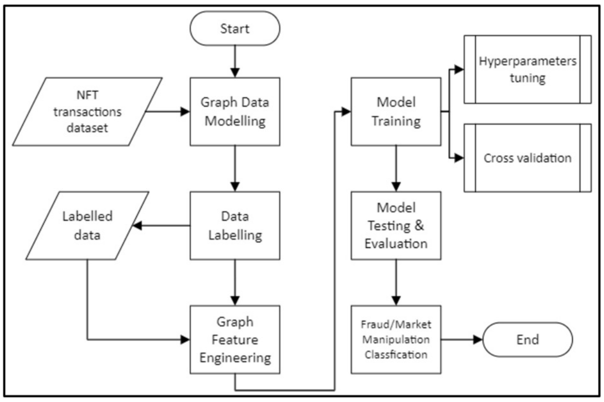
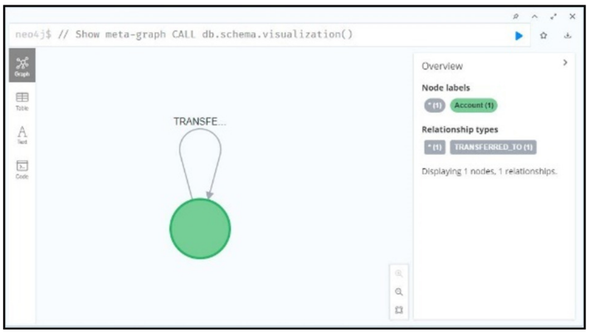


Fig.1: *Research methodology.*

3.1 Data Labelling

The graph representation of the labelling process is a homogeneous graph with one node type (accounts) and one edge type (transactions). In addition to the labelling process, graph representation extracts relevant features for inclusion in the model.

Figure 2 shows a Neo4j application that displays a homogeneous graph schema. The transaction data per NFT object taken to be labelled are those for objects with at least seven trading activities on the related object, because this is the minimum number of wash-trading patterns that can be observed [15]. The criterion yielded 7,351 NFT objects with at least seven and up to 272 buying and selling transactions for each item.



```
neo4j$ // Show meta-graph CALL db.schema.visualization()
```

Overview

Node labels

*(1) Account (1)

Relationship types

*(1) TRANSFERRED_TO (1)

Displaying 1 nodes, 1 relationships.

Fig.2: *Graph scheme for the data labelling process.*

According to several studies [28], [29], closed-loop transaction patterns can identify wash trading operations. Specific algorithms can quickly identify these patterns in data represented as graphs [29]. Consequently, this study examined graph-modelled NFT transaction data. Graph modelling is an excellent tool for representing data structures linked in a blockchain network [43]. A large-scale graph model was used as input to a neural network, a graph neural network (GNN), for classification.

However, the absence of a dataset containing NFT transaction data flagged as fraudulent hinders our investigation. Nevertheless, existing market manipulation studies [15], [29] provided strategies for identifying transaction patterns that suggest fraud. Therefore, a data labelling process is performed on existing datasets by detecting a closed cycle/ring topology pattern that indicates wash trading activity [15], [29] using the Depth-First Search (DFS) algorithm.

The data labelling process follows the graph analysis method used to detect suspicious transactions in NFT trading [29], namely, checking for closed loops in a series of transactions for each NFT. A closed loop occurs when an NFT changes ownership multiple times through a series of buying and selling actions, eventually returning to the original owner's hands. According to the study, a closed-loop transaction indicating fraud occurs when it occurs within 12 hours.

Further data processing shows that the prices of NFT objects in transactions labelled as fraud form a pattern that does not fully follow Benford's Law. Benford's Law, also known as the first-digit Law (or Law of anomalous numbers), states that in an observation, number 1 as the first digit of a number should appear in at least 30% of the observations, and consecutively until number 9 appears at least, or approximately 5% [44]. According to this Law, if a quantitative number observation does not follow this pattern, it indicates an anomaly or fraud within the data.

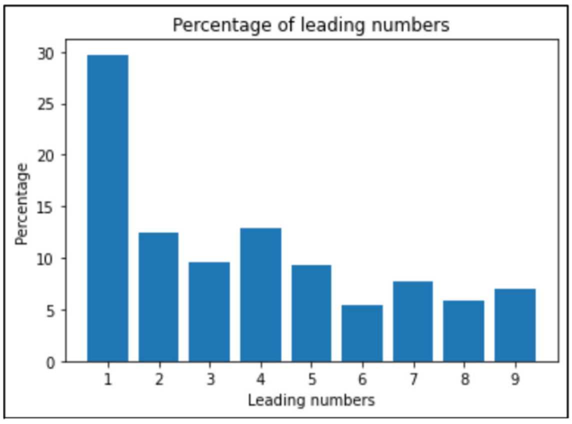


Fig.3: *Distribution of leading digits according to Benford's Law.*

An example of a suspected market manipulation transaction is shown in Figure 4. This indication of market manipulation is evident across two main clusters: the upper cluster, centred on an address such as 0×9428E, shows complex, circular, and frequent transfers between multiple wallets to inflate apparent trading volume artificially. Concurrently, the lower cluster, particularly the dense network between 0×5b411 and 0xA7311, exhibits an extreme concentration of "TRANSFERRED_TO" arrows, demonstrating that the NFT was rapidly transferred back and forth multiple times within a short period. This combined activity serves to manipulate an asset's perceived value and price data by fabricating a false record of legitimate sales, ultimately constituting a

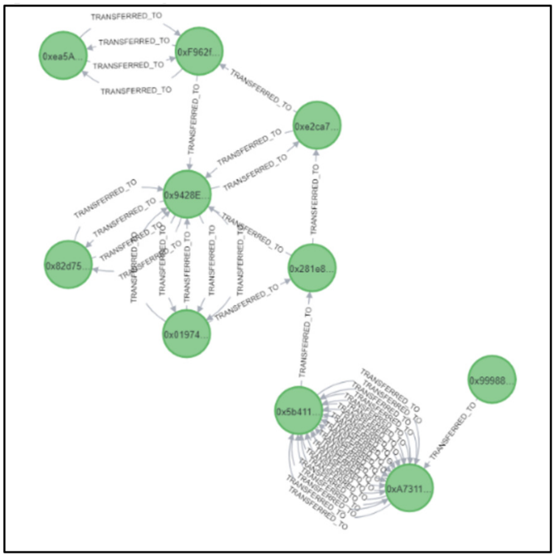


Fig.4: *The transaction pattern of an NFT object from the dataset is indicated as fraud.*

deceptive market practice.

The data labelling process identified 36,177 transactions suspected of being part of market manipulation. This number is insignificant compared to the total number of transactions in the dataset, representing only 0.08%. A sampling strategy can overcome this imbalance. The B sampling strategy can balance the classes in the classification model. Two sampling strategies can address imbalances: oversampling to increase the number of minority classes relative to the majority, and undersampling to reduce the number of majority classes relative to minorities.

The sampling strategy in this study used oversampling and undersampling to balance the data between the two classes. We executed the test with three ratio scenarios: 0.2, 0.4, and 0.6. The ratio is the proportion of new data in each class relative to the total amount of data. A ratio of 0.4 means that the fraud and non-fraud classes account for 40% of the initial data.

Table 1: *Number of entities in each sampling strategy scenario.*

	0,2	0,4	0,6
Account	659.916	809.108	899.054
NFT	1.970.860	3.337.490	4.353.536
Transaction	1.791.420	3.582.840	5.374.262

Table 1 lists the number of entities in each node type for each sampling strategy ratio scenario.

3.2 Graph Feature Engineering

The next step after labelling the data was to extract the relevant features. The process of extracting this feature involves three graph algorithms to determine the properties of the graph representation: PageRank, degree centrality, and greedy modularity communities.

3.3 PageRank Algorithm

The PageRank algorithm is used in one of the network properties, link analysis, to determine the node attributes. According to its original premise, the PageRank algorithm helps determine a webpage's relevance by counting the number of links pointing to it from other pages [45]. The greater the number of links that point to a webpage, the more critical the webpage is; hence, the higher its PageRank score. In general, the PageRank algorithm is calculated using the following equation:

$$PR(u) = \sum\nolimits_{v \in B_u} \frac{PR(v)}{L(v)} \tag{1}$$

Where $PR(u)$ is the PageRank score of page u, which is calculated as the sum of the PageRank values of page v (pages that point to page u) divided by the number of links from those coming from page v. The starting value of the PageRank is determined proportionally based on the initial number of web pages counted.

In the context of market manipulation, a party engaging in wash trading typically comprises several individuals who frequently design buying and selling transactions to create the illusion of spontaneous transactions. As a result, several transactions led to these individuals. Consequently, these individuals typically have better PageRank scores than other accounts. However, high-scoring accounts are not always involved in market manipulation.

3.4 Degree Centrality Algorithm

In a graph network, the Degree Centrality algorithm measures the total number of links pointing directly to a node [46]. In other words, a degree is a measure of the risk that a node will incur because of its connection to other nodes. The degree centrality score of a vertex or node of a graph $G := (V, E)$, where $|V|$ is the number of vertices and $|E|$ represents the edges, is calculated as follows:

$$C_D(v) = \deg(v) \tag{2}$$

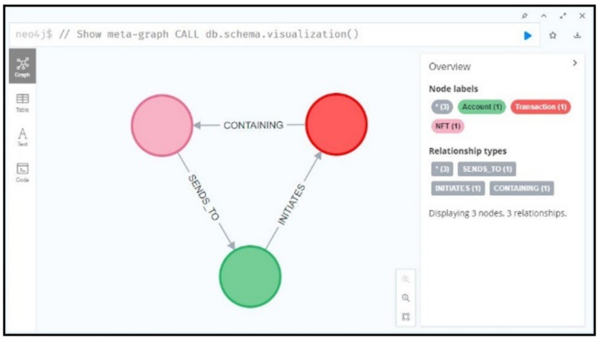
In the wash trading context, the entity that devises the wash trading plan is usually at the heart of the engineered transactions. As a result, these individuals typically have a higher degree centrality score than other accounts.

3.5 Greedy Modularity Algorithm

The last algorithm used was Greedy Modularity Communities. This algorithm can help identify a crucial property of a graph: communities. The greedy modularity community algorithm, commonly known as the Girvan-Newman algorithm, partitions a graph's nodes into communities [47]. In the context of market manipulation, this algorithm groups accounts that collude to create engineered transactions. Because these accounts work together to engineer transactions, when the data are modeled as a graph, the algorithm classifies the accounts into the same community.

3.6 Graph Data Modeling

Another graph representation was created after labeling the data and extracting features as inputs for testing the Graph Neural Network classification model. The graph representation is based on the same data as the preceding graph; however, it is heterogeneous, meaning it has multiple node and edge types. A schematic representation of this graph is shown in Figure 3.



```
neo4j$ // Show meta-graph CALL db.schema.visualization()
```

Displaying 3 nodes, 3 relationships.

Fig.5: *Graph scheme for the classification model input.*

All transaction rows are converted into graph nodes to create the transaction nodes. Account nodes are produced by merging account data from the _address and to address columns, which are then combined to provide unique account data after the removal of duplicate data. Meanwhile, NFT nodes are constructed by querying all transaction rows and grouping them by nft_address and token_id. This graph representation contains 4,514,729 transaction nodes, 509,577 account nodes, and 2,860,215 NFT nodes in total.

3.7 Model Training and Testing

This study includes two main test scenarios: testing a model with a graph representation that already includes algorithm scores, and evaluating a model without these features. Three deep learning models were used to test these two scenarios: Multilayer Perceptron (MLP), Graph Convolutional Neural Network (GCN), and heterogeneous Graph Neural Network (HeteroGCN).

3.8 Hyperparameters Tuning

Several factors, including the sampling technique, learning rate, dropout, and weight decay, were examined during hyperparameter tuning. To counteract uneven data, the sampling approach employed over- and undersampling. The learning rate is a hyperparameter that controls how quickly the model updates its weights based on its predictions and the data it trains on. Dropout is a strategy for improving model accuracy. This hyperparameter prevents potential overfitting by deactivating a small percentage of hidden units during training. Weight decay is another hyperparameter that can help reduce overfitting. The sampling method is a hyperparameter tested across all models; however, the learning rate and weight decay are tested only on the MLP and Heterogeneous GCN models, and dropout is tested only on GCN models.

Table 2: *Candidates of hyperparameters to be tested in model testing; bold values are default values.*

Hyperparameters	**Candidates**
Sampling strategy ratio	[**0.2**, 0.4, 0.6]
Learning rate	[0.01, **0.05**, 0.10]
Dropout	[0.25, **0.5**, 0.75]
Weight decay	[0.0005, **0.001**, 0.005]

3.9 Cross-validation

The validation process must employ a statistical method to estimate the model's predictive ability. Each training/testing process performed by the model in this study was validated using k-fold cross-validation. The approach randomises the data before dividing them into k equal halves, with k = 10. The initial iteration used nine parts of the data for training and 1 part for testing. The following iteration used nine different parts for training and one for testing.

4. RESULTS AND DISCUSSION

The results of the sampling strategy test using the HeteroGCN model with the three predetermined scenarios were identical, as shown in Table 2. Therefore, given the time, memory, and computational resources required for model testing, a sampling ratio of 0.2 was selected to address data imbalance.

Table 3: *Sampling strategy test results.*

Sampling strategy	F1 Score	Accuracy	Recall	Precision
Ratio 0.2	0,994	0,994	0,997	0,990
Ratio 0.4	0,994	0,994	0,997	0,990
Ratio 0.6	0,994	0,994	0,997	0,990

With three models and two scenarios tested, six experiments were conducted: the MLP model without features (MLP-NF), GCN model without features (GCN-NF), HGCN model without features (HGCN-NF), MLP model with features (MLP-F), GCN model with features (GCN-F), and HGCN model with features (HGCN-F). Each tested model underwent validation using K-fold cross-validation.

Table 4: *Test result metrics summary.*

Model scenarios	*F1 Score*	*Accuracy*	*Recall*	*Precision*
MLP-NF	0,0	0,5	0,0	0,0
MLP-F	0,992	0,992	0,994	0,990
GCN-NF	0,0	0,5	0,0	0,0
GCN-F	0,974	0,974	0,970	0,977
HGCN-NF	0,0	0,5	0,0	0,0
HGCN-F	0,994	0,994	0,997	0,990

There is a stark contrast between the models that include scores calculated by the graph algorithm (as shown in Tables 3 and 4), because models without these additional features cannot minimise the loss function to its minimum, as shown in Figure 6. Figure 6 shows a critical finding regarding the models trained without additional graph-based features. The loss functions for all three models (MLP, GCN, and HeteroGCN) stagnate at approximately 0.69 after an initial small drop. This plateau indicates that the models failed to learn meaningful patterns from the data because they were unable to minimise their prediction errors during training.

This failure to learn directly results in poor performance, as illustrated by the ROC curve in Figure 7. The ROC curve for the MLP model without features is a straight diagonal line, indicating a classifier with no predictive power —equivalent to random guessing (AUC = 0.5). This visually confirms the quantitative results in Table 4, where the models without features (MLP-NF, GCN-NF, HGCN-NF) achieved accuracies of only 0.5 and F1-scores of 0.0. In stark contrast, the models enhanced with additional features achieved near-perfect classification, highlighting the importance of these features for the model's success.

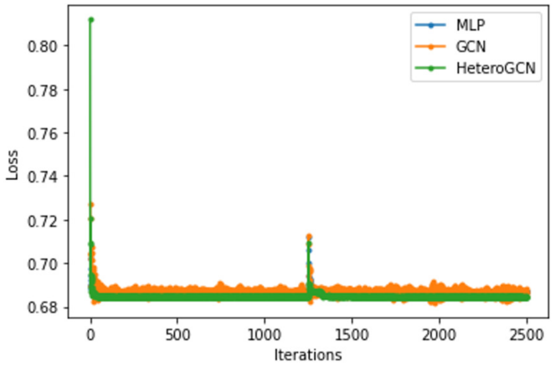


Fig. 6: *Loss function calculation results of each iteration for models with no extra features.*

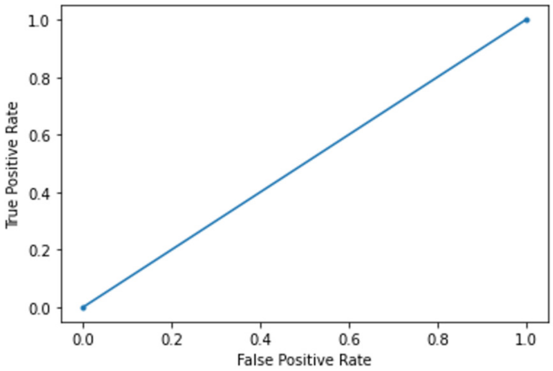


Fig. 7: *ROC curve of the MLP model with no extra features.*

The difference in performance between models with and without additional features was also evident in the confusion matrix. A confusion matrix was used to assess classification quality, recording both true and false instances for each class [48]. In this study, models lacking additional features failed to group samples within each class.

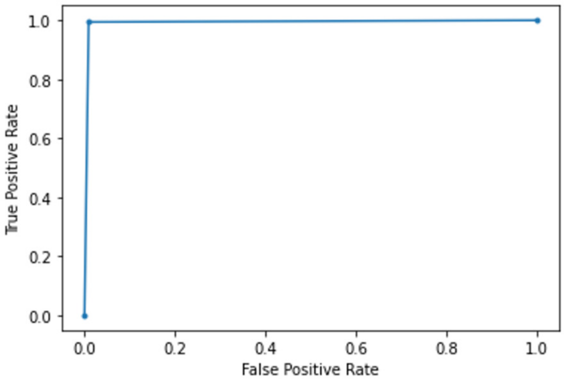


Fig. 8: *ROC curve of the MLP model with extra features.*

The area under the ROC curve in Figure 7 (0.5), indicating that it is no better than a random classifier, corresponds to the findings of the confusion matrix for models without these additional features.

In stark contrast to the model performance without additional features, Figure 8 shows the ROC curve for the MLP model enhanced with these features. The curve demonstrates near-perfect classification, as it rises vertically to a True Positive Rate of 1.0 while maintaining a False Positive Rate of 0.0.

This ideal shape, which occupies the top-left corner of the plot, corresponds to an Area Under the Curve (AUC) of 1.0, which is the maximum possible score. This confirmed that the model was highly discriminative and effective. This outstanding performance is a direct result of incorporating extra features, which enabled the model to learn the underlying data patterns successfully, as further quantified by the high precision and recall values shown in the confusion matrix in Table 5.

Table 5: *Confusion matrix test results of models that do not have additional features (top) and that have extra features (bottom).*

	Prediction +	Prediction -	
Actual +	895.710	0	895.710
Actual -	895.710	0	895.710
	1.791.420	0	1.791.420
	Prediction +	Prediction -	
Actual +	887.353	8.357	895.710
Actual -	2.220	893.490	895.710
	889.573	901.847	1.791.420

Another example is the performance of models with additional features. In this scenario (Table 5), the model correctly predicted the classes in 887,353 samples from one class out of 895,710 (99% accuracy) and in 893,490 samples from the other classes (99.7% accuracy). This explains why the ROC curves of the models with the added perfect classification characteristic were prominent (Figure 6).

The model developed in this study significantly reduces the risks for all parties involved. For investors, the system minimises the risk of undetected False Negatives (FN) fraud, preventing manipulators from artificially inflating NFT prices through wash trading and subsequent dumping, thereby protecting investors from financial losses. For regulators, this low FN rate means fewer instances of illegal market manipulation go unpunished, thereby upholding market integrity. Crucially, the model also shows an extremely low False Positive (FP) rate, resulting in a high precision score. This low FP rate is essential for marketplaces and legitimate users, as it prevents the system from incorrectly flagging or freezing honest trading accounts, thereby preserving the user experience and the platform's reputation. Overall, the model is exact and successfully captures nearly all actual fraud cases, prioritising the reduction of risks associated with undetected manipulation.

To compensate for the data imbalance, these models were tested using over- and undersampling. The classification models examined in this stage used graph algorithm scores as inputs.

Table 6: *Report on classification based on the Random Forest algorithm.*

	Precision	Recall	F1-score	Support
Non-fraud	0,98	0,98	0,98	358.585
Fraud	0,98	0,98	0,98	357.983
Accuracy			0,98	716.568
Macro avg	0,98	0,98	0,98	716.568
Weighted avg	0,98	0,98	0,98	716.568
Train accuracy			0,999	
Test accuracy			0,981	

Table 7: *Report on classification based on a model that employs the Decision Tree algorithm.*

	Precision	Recall	F1-score	Support
Non-fraud	0,98	0,98	0,98	895.600
Fraud	0,98	0,98	0,98	895.821
Accuracy			0,98	1.791.421
Macro avg	0,98	0,98	0,98	1.791.421
Weighted avg	0,98	0,98	0,98	1.791.421
Train accuracy			0,999	
Test accuracy			0,978	

Table 6 shows the classification report after applying the Random Forest technique to evaluate the classification model. The categorisation model achieved 98% accuracy according to the test results.

Table 8: *Report on classification based on the K-Nearest Neighbour algorithm.*

	Precision	Recall	F1-score	Support
Non-fraud	0,97	0,94	0,96	357.951
Fraud	0,94	0,97	0,96	358.617
Accuracy			0,96	716.568
Macro avg	0,96	0,96	0,96	716.568
Weighted avg	0,96	0,96	0,96	716.568
Train accuracy			0,969	
Test accuracy			0,956	

Meanwhile, Table 7 shows that the classification model utilising the Decision Tree approach achieved an accuracy of 97,8%. In contrast, Table 8 shows that the classification model using the K-Nearest Neighbour technique attained an accuracy of 95,6%.

The classification report indicates that deep learning-based models outperform machine learning-based models. However, these advantages are marginal because the accuracy scores of each model varied only slightly. This observation was corroborated by the overlay of the ROC curve of each model (Figure 9).

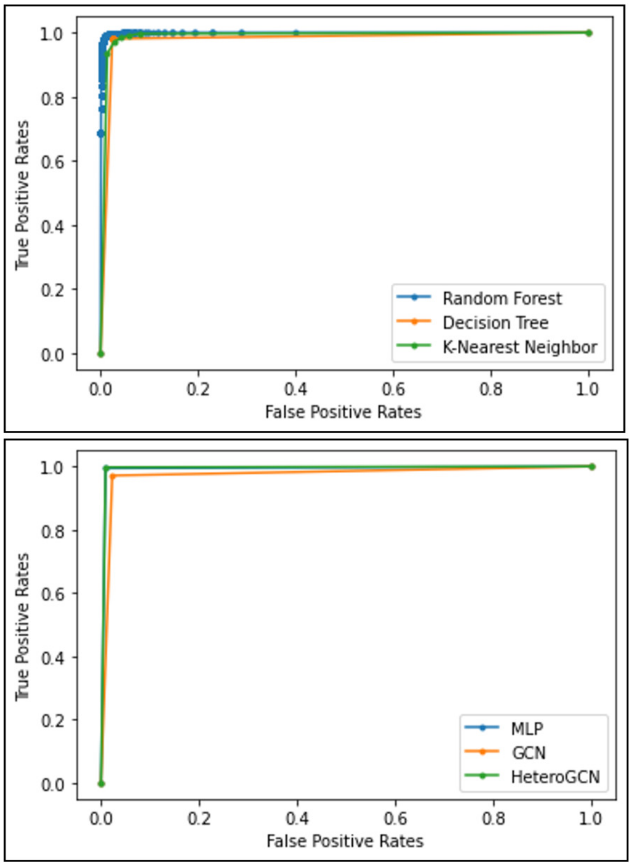


Fig. 9: *ROC curves of each model based on machine learning and deep learning.*

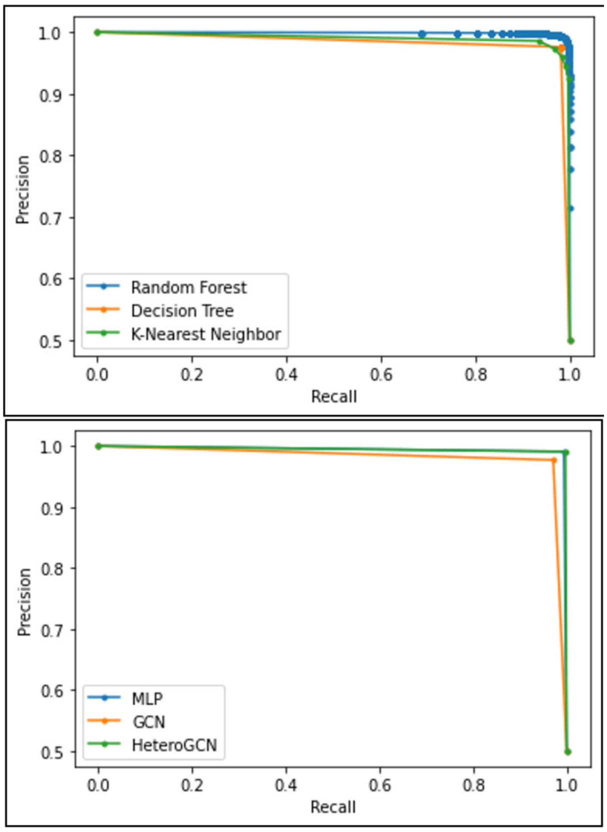


Fig. 10: *The PR curve compares each recall score threshold to the average precision score.*

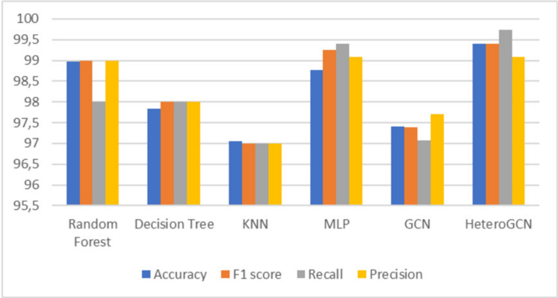


Fig. 11: *Comparison chart between accuracy, F1 score, recall, and precision values for each machine learning and deep learning-based classification model.*

The area under the precision (AUPR) curve exhibited the same pattern. The AUPR curve, similar to the ROC curve, measures model performance. This curve combines precision and recall on the Y- and X-axes, respectively. The higher the curve on the Y-axis, the better the performance of the model.

A more detailed summary of the metrics for comparing classification model groups based on machine learning and deep learning is shown in Figure 9.

Figure 11 shows that the HeteroGCN model outperformed the categorisation models. The model achieved an F1 Score of 0.994, an accuracy score of 0.994, a recall score of 0.997, and a precision score of 0.99.

The MLP model was the second-best performer, with an F1 score of 0.992, accuracy of 0.992, recall of 0.994, and precision of 0.990. Notably, GCN models perform poorly even when compared with machine-learning-based classification models. This may be because the GCN model is a Graph Neural Network that takes homogeneous graphs as input, whereas this study uses heterogeneous graphs.

Furthermore, the HeteroGCN model was used to identify which features significantly affected the classification performance of the grouping model for fraudulent and non-fraudulent transactions. Six scenarios were prepared for testing.

- A model with Degree Centrality (DC) algorithm score features only,
- A model with PageRank algorithm score features only (PR),
- A model with the Greedy Modularity Communities algorithm score features only (GMC),
- A model with PageRank and Greedy Modularity Communities algorithm score features (PR-GMC),
- A model with PageRank and Degree Centrality algorithm score features (PR-DC),
- A model with degree centrality and greedy modularity community algorithm score features (DC-GMC).

The test results of each scenario are summarised in Table 9

Table 9: *Summary of metrics on backtesting for feature significance.*

Scenario	Accuracy	F1 Score	Recall	Precision
DC	0,994	0,994	0,997	0,990
PR	0,992	0,992	0,994	0,991
GMC	0,5	0,0	0,0	0,0
PR-GMC	0,992	0,992	0,993	0,990
PR-DC	0,994	0,994	0,997	0,990
GMC-DC	0,994	0,994	0,997	0,990

Table 9 summarises metrics indicating that feature pair combinations perform well at predicting transaction classes. When testing the model with features as single scores from each algorithm, models with Degree Centrality and PageRank algorithm scores predicted transaction classification well, whereas models with greedy modularity community algorithm scores did not. Graph metrics, specifically PageRank (PR) and Degree Centrality (DC), are effective for identifying accounts involved in wash trading, as central manipulator accounts accumulate high scores through orchestrated circular transactions and numerous interactions with dummy wallets. In contrast, the Greedy Modularity Communities (GMC) algorithm performed poorly as a standalone feature, resulting in a model accuracy of 0.5 and an F1 score of 0.0. The decentralized, unregulated nature of account creation on blockchain platforms directly causes this deficiency, as it allows manipulators to create an excessive number of controlled addresses. This spreading out of shell accounts hides the actual network structure, preventing the GMC algorithm from reliably distinguishing between legitimate trading communities and artificial clusters formed by a single colluding entity executing ring topology transactions.

The poor model performance with the GMC algorithm score characteristics suggests that community attributes in a graph of buyer/seller accounts that transact NFTs do not play a role in assessing whether a transaction is fraudulent. This lack of distinction is possible because the model cannot distinguish between accounts that conspire to manipulate NFT transactions and those that engage in legitimate NFT transactions. Because there are no constraints on this in a blockchain network, a person or group can create an unlimited number of accounts to mask their identities.

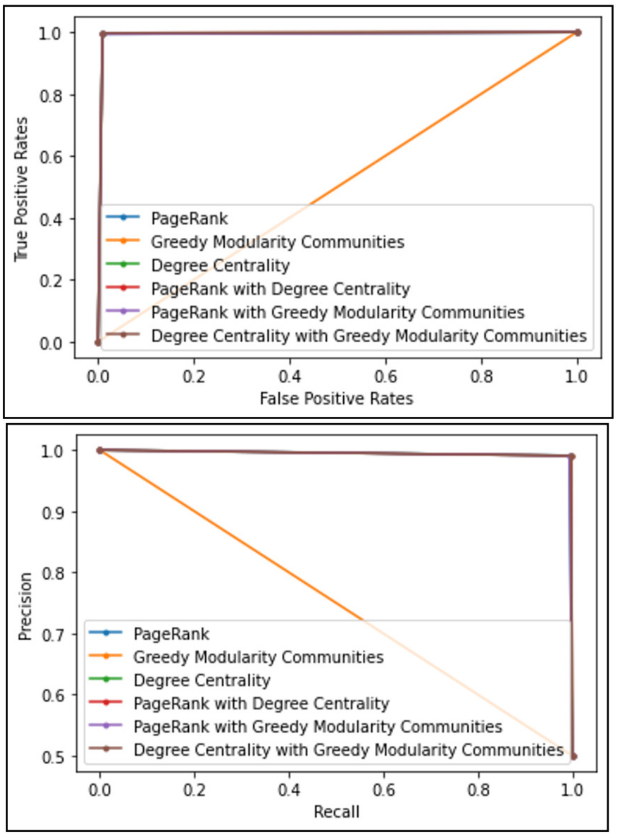


Fig.12: *ROC (top) and PR (bottom) curves from model testing on several feature scenarios.*

5. CONCLUSION

This study demonstrates how to use a graph-modelled dataset to build a classification model that predicts market manipulation behaviour in NFT transactions. Wash trading —the act of purchasing and selling NFTs manufactured by a person or group to give the impression that a digital asset has a high selling value —is the NFT market manipulation activity on which this study focuses.

This study further shows that adding characteristics to the input data, in the form of scores from the computation of several graph algorithms, can significantly enhance model performance. The test results suggest that the PageRank and Degree Centrality algorithms improved the classification model's predictive performance.

Furthermore, the detection system has significant regulatory implications, as its ability to identify suspicious, highly interconnected accounts helps financial institutions and regulatory bodies meet Anti-Money Laundering (AML) requirements by targeting wallets involved in schemes that create the illusion of active market volume. To fully confirm the generality and robustness of the Graph Neural Network (GNN) methodology, future work must focus on two key areas. First, the detection model must be extended to assets on multiple blockchain networks to validate its performance across diverse environments. Second, it is crucial to validate the model using extended time ranges of transaction data to mitigate temporal bias, a significant challenge in the continuous detection of crypto fraud.

ACKNOWLEDGEMENT

This work would not be possible without financial support from the Information Systems Department, Institut Teknologi Sepuluh Nopember, Surabaya, in accordance with Research Implementation Agreement Letter No: 1821/PKS/ITS/2022.

AUTHOR CONTRIBUTIONS

Conceptualisation, Ade Indriawan and Nur Aini Rakhmawati; methodology, Nur Aini Rakhmawati; software, Ade Indriawan; validation, Nur Aini Rakhmawati; formal analysis, Nur Aini Rakhmawati; investigation, Ade Indriawan; data curation, Ade Indriawan; writing—original draft preparation, Ade Indriawan; writing—review and editing, Ade Indriawan and Nur Aini Rakhmawati; visualisation, Ade Indriawan; supervision, Nur Aini Rakhmawati; funding acquisition, Nur Aini Rakhmawati. All the authors have read and agreed to the published version of the manuscript.

References

[1] D. Chalmers, C. Fisch, R. Matthews, W. Quinn and J. Recker, "Beyond the bubble: Will NFTs and digital proof of ownership empower creative industry entrepreneurs?," *Journal of Business Venturing Insights*, vol. 17, p. e00309, Jun. 2022.

[2] E. A. Akartuna, S. D. Johnson and A. Thornton, "Preventing the money laundering and terrorist financing risks of emerging technologies: An international policy Delphi study," *Technological Forecasting and Social Change*, vol. 179, p. 121632, Jun. 2022.

[3] L. Almaqableh *et al.*, "Is it possible to establish the link between drug busts and the cryptocurrency market? Yes, we can," *International Journal of Information Management*, p. 102488, Feb. 2022.

[4] J. T. Hamrick *et al.*, "An examination of the cryptocurrency pump-and-dump ecosystem," *Information Processing & Management*, vol. 58, no. 4, p. 102506, Jul. 2021.

[5] H. Nghiem, G. Muric, F. Morstatter and E. Ferrara, "Detecting cryptocurrency pump-and-dump frauds using market and social signals," *Expert Systems with Applications*, vol. 182, p. 115284, Nov. 2021.

[6] M. Bartoletti, S. Carta, T. Cimoli and R. Saia, "Dissecting Ponzi schemes on Ethereum: Identification, analysis, and impact,"*Future Generation Computer Systems*, vol. 102, pp. 259–277, Jan. 2020.

[7] S. Fan, S. Fu, H. Xu and C. Zhu, "Expose Your Mask: Smart Ponzi Schemes Detection on Blockchain," *2020 International Joint Conference on Neural Networks (IJCNN)*, Glasgow, UK, pp. 1-7, 2020.

[8] G. Y. Pratomo, "Viral NFT Ghozali, Beri Potensi Perkembangan Aset Digital di Indonesia - Crypto Liputan6.com," 2022. https://www.liputan6.com/crypto/read/4860112/viral-nft-ghozali-beri-potensi-perkembangan-aset-digital-di-indonesia (accessed Apr. 19, 2022).

[9] CNN Indonesia, "Survei: Tren NFT 51 di RI Bakal Berlangsung 5 Tahun ke Depan," 2022. https://www.cnnindonesia.com/teknologi/20220215133220-185-759390/survei-tren-nft-51-di-ri-bakal-berlangsung-5-tahun-ke-depan (accessed Apr. 19, 2022).

[10] K. Runiasari, "Latah artis bikin koin kripto: Sekadar aji mumpung?," 2022. https://www.alinea.id/bisnis/latah-artis-bikin-koin-kripto-sekadar-aji-mumpung-b2fgK9ClU (accessed Apr. 19, 2022).

[11] M. Yefriza, "Begini Nasib Token ASIX Anang dan I-COIN Wirda Mansur — Tagar," 2022. https://www.tagar.id/begini-nasib-token-asix-anang-dan-icoin-wirda-mansur (accessed April 19, 2022).

[12] S. Imisiker and B. K. O. Tas, "Wash trades as a stock market manipulation tool," *Journal of Behavioral and Experimental Finance*, vol. 20, pp. 92–98, Dec. 2018.

[13] C. McKhann, "What Is A Short Squeeze And What Is Going On In GameStop, AMC," INVESTOR'S CORNER, 2022. https://www.investors.com/how-to-invest/investors-corner/short-squeeze/ (accessed June 12, 2023).

[14] Y. Cao, Y. Li, S. Coleman, A. Belatreche and T. M. McGinnity, "Detecting Wash Trade in Financial Market Using Digraphs and Dynamic Programming," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 11, pp. 2351-2363, Nov. 2016.

[15] Y. Cao, Y. Li, S. Coleman, A. Belatreche and T. M. McGinnity, "Detecting wash trade in the financial market," *2014 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr)*, London, UK, pp. 85-91, 2014.

[16] X. Huang, L. Wu and Y. Ye, "A Review on Dimensionality Reduction Techniques," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 33, no. 10, p. 1950017, Sep. 2019.

[17] W. Jia, · Meili Sun, J. Lian and S. Hou, "Feature dimensionality reduction: a review," *Complex & Intelligent Systems* , vol. 8, pp. 2663–2693, 2022.

[18] C. Pirrong, "The economics of commodity market manipulation: A survey,"*Journal of Commodity Markets*, vol. 5, pp. 1–17, Mar. 2017.

[19] F. Allen and D. Gale, “Stock-Price Manipulation,” *The Review of Financial Studies*, vol. 5, no. 3, pp. 503–529, Jul. 1992.

[20] J. Xu and B. Livshits, “The Anatomy of a Cryptocurrency Pump-and-Dump Scheme,” *Proceedings of the 28th USENIX Conference on Security Symposium*, 2019, Accessed: May 23, 2022. [Online]. Available: `www.usenix.org/conference/usenixsecurity19/presentation/xu-jiahua`

[21] J. Wu *et al.*, “Who Are the Phishers? Phishing Scam Detection on Ethereum via Network Embedding,” in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 2, pp. 1156-1166, Feb. 2022.

[22] S. A. Tariq and I. Sifat, “Suspicious Trading in Non-fungible Tokens (Nfts): Evidence from Wash Trading,” *SSRN Electron. J.*, May 2022.

[23] W. Chen, Z. Zheng, J. Cui, E. Ngai, P. Zheng and Y. Zhou, “Detecting Ponzi schemes on Ethereum: Towards healthier blockchain technology,” *Web Conf. 2018 - Proceedings of the 2018 World Wide Web Conference (WWW 2018)*, vol. 4, pp. 1409–1418, Apr. 2018.

[24] M. Weber *et al.*, “Anti-Money Laundering in Bitcoin: Experimenting with Graph Convolutional Networks for Financial Forensics,” *arXiv preprint* arXiv:1908.02591 Jul. 2019.

[25] V. Chandola, A. Banerjee and V. Kumar, “Anomaly detection,”*ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, Jul. 2009.

[26] S. Li, “Anomaly Detection for Dummies,” 2019. `https://towardsdatascience.com/anomaly-detection-for-dummies-15f148e559c1` (accessed June 19, 2022).

[27] K. Golmohammadi, O. R. Zaiane and D. Díaz, “Detecting stock market manipulation using supervised learning algorithms,” *2014 International Conference on Data Science and Advanced Analytics (DSAA)*, Shanghai, China, pp. 435-441, 2014.

[28] S. Serneels, “Detecting wash trading for non-fungible tokens,” *Finance Research Letters*, vol. 52, p. 103374, Mar. 2023.

[29] V. von Wachter, J. R. Jensen, F. Regner and O. Ross, “NFT Wash Trading: Quantifying Suspicious Behaviour in NFT markets,” *Financial Cryptography and Data Security. FC 2022 International Workshops: CoDecFin, DeFi, Voting, WTSC, Grenada*, pp. 299-311, 2022.

[30] G. Pang, C. Shen, L. Cao and A. Van Den Hengel, “Deep Learning for Anomaly Detection,” *ACM Computing Surveys*, vol. 54, no. 2, Mar. 2021..

[31] A. Li, J. Wu, and Z. Liu, “Market Manipulation Detection Based on Classification Methods,” *Procedia Computer Science*, vol. 122, pp. 788–795, 2017.

[32] Q. Liu, C. Wang, P. Zhang and K. Zheng, “Detecting stock market manipulation via machine learning: Evidence from China Securities Regulatory Commission punishment cases,” *International Review of Financial Analysis*, vol. 78, p. 101887, Nov. 2021.

[33] J. Lorenz, M. I. Silva, D. Aparício, J. T. Ascensão and P. Bizarro, “Machine learning methods to detect money laundering in the bitcoin blockchain in the presence of label scarcity,” in *Proceedings of the First ACM International Conference on AI in Finance*, pp. 1–8, Oct. 2020.

[34] L. Chen, J. Peng, Y. Liu, J. Li, F. Xie and Z. Zheng, “Phishing Scams Detection in Ethereum Transaction Network,” *ACM Transactions on Internet Technolog*, vol. 21, no. 1, pp. 1–16, 2020.

[35] A. Singh, A. Gupta, H. Wadhwa, S. Asthana and A. Arora, “Temporal Debiasing using Adversarial Loss based GNN architecture for Crypto Fraud Detection,” *2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Pasadena, CA, USA, pp. 391-396, 2021.

[36] R. Tan, Q. Tan, P. Zhang and Z. Li, “Graph Neural Network for Ethereum Fraud Detection,” *2021 IEEE International Conference on Big Knowledge (ICBK)*, Auckland, New Zealand, pp. 78-85, 2021.

[37] L. Xie, D. Pi, X. Zhang, J. Chen, Y. Luo, and W. Yu, “Graph neural network approach for anomaly detection,” *Measurement*, vol. 180, p. 109546, Aug. 2021.

[38] A. Hodler and M. Needham, *Graph Data Science for Dummies*. New Jersey, 2021.

[39] M. E. J. Newman, “The Structure and Function of Complex Networks *,” 2003. [Online]. Available: http://www.siam.org/journals/sirev/45-2/42480.html

[40] A. Gupta, P. Matta, and B. Pant, “Graph neural network: Current state of Art, challenges and applications,” *Materials Today: Proceedings*, vol. 46, pp. 10927–10932, Jan. 2021.

[41] J. Zhou *et al.*, “Graph Neural Networks: A Review of Methods and Applications,” *AI Open*, vol. 1, pp. 57–81, Dec. 2020.

[42] H. Kanezashi, T. Suzumura, X. Liu and T. Hirofuchi, “Ethereum Fraud Detection with Heterogeneous Graph Neural Networks,” *arXiv preprint* arXiv:2203.12363 , Mar. 2022.

[43] A. Khan, “Graph Analysis of the Ethereum Blockchain Data: A Survey of Datasets, Methods, and Future Work,” *2022 IEEE International Conference on Blockchain (Blockchain)*, Espoo, Finland, pp. 250-257, 2022.

[44] A. Berger and T. P. Hill, “Benford’s Law Strikes Back: No Simple Explanation in Sight for Mathematical Gem,” *The Mathematical Intelligencer*, vol. 33, pp. 85-91, 2011.

[45] L. Page, S. Brin, R. Motwani and T. Wino-

grad, “The PageRank Citation Ranking: Bringing Order to the Web,” 1998, [Online]. Available: `http://ilpubs.stanford.edu/422/1/1999-66.pdf`

[46] J. Zhang and Y. Luo, “Degree Centrality, Betweenness Centrality, and Closeness Centrality in Social Network,” in *2nd International Conference on Modelling, Simulation and Applied Mathematics (MSAM 2017)*, vol. 132, pp. 300-303, 2017.

[47] M. E. J. Newman and M. Girvan, “Finding and evaluating community structure in networks,” Physical Review E - Stat. Nonlinear, Soft Matter Phys., vol. 69, no. 2, Aug. 2004.

[48] A. Li, J. Wu and Z. Liu, “Market Manipulation Detection Based on Classification Methods,” *Procedia Computer Science*, vol. 122, pp. 788–795, Jan. 2017.

Ade Indriawan is affiliated with the Institut Teknologi Sepuluh Nopember (ITS), Surabaya, and Imperiotica Strategic Lab. He holds a Master’s Degree in Information Systems from ITS, where his academic and research interests focus on knowledge graphs, machine learning, deep learning, graph neural networks, and blockchain technologies. His previous publications include “*Characteristics of Blockchain-based Digital Asset Datasets: A Systematic Review*” presented at the 6th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE 2022), and “*Movies Analysis on DBpedia and Wikidata Using Community Detection and Centrality Algorithms*” presented at the International Electronics Symposium (IES 2022). He was also recognized as a Champion of INOVBOYO 2022, Surabaya’s city innovation contest, for developing Surabaya Halal Directory, a machine learning project that maps halal-certified food vendors across the city.

Nur Aini Rakhmawati is a professor in the Information Systems Department, Institut Teknologi Sepuluh Nopember (ITS) in Surabaya, Indonesia. She earned her PhD from the Insight Centre for Data Analytics at NUI Galway, Ireland, her Master’s degree from the National Taiwan University of Science and Technology, and her Bachelor’s degree from ITS Surabaya. Her current research interests include knowledge graphs, big data, and computer ethics.