It's All Connected: Detecting Phishing Transaction Records on Ethereum Using Link Prediction

Chidimma Opara[1], Yingke Chen[2], and Bo Wei[3]

[1] Teesside University, Middlesbrough, UK
[2] Northumbria University, Newcastle Upon Tyne, UK
[3] Lancaster University, Lancaster, UK

Abstract. Smart contracts are increasingly being used on platforms for virtual transactions, such as Ethereum, owing to new financial innovations. As these platforms are anonymous and easy to use, they are perfect places for phishing scams to grow. Unlike traditional phishing detection approaches that aim to distinguish phishing websites and emails using their HTML content and URL, phishing attacks on Ethereum focus on detecting phishing addresses by analysing the transaction relationships on the virtual transaction platform. This study proposes a link prediction framework for detecting phishing transactions on the Ethereum platform using 12 local network-based features extracted from the ether receiving (target) and initiating (source) addresses. The framework was trained and tested on over 280,000 verified phishing and legitimate transaction records. Experimental results indicate that the proposed framework with a LightGBM classifier provides a high recall of 89% and an AUC score of 93%.

Keywords: Phishing detection, Ethereum Network, Link prediction, Graph representation.

1 Introduction

Since the introduction of blockchain, a distributed ledger, in 2008, it has captured the attention of the industry and academia. The ledger records the number of users' cryptocurrency and the history of transfer transactions between them. The user is represented in the system as a public-private key pair. Public keys, often called addresses, are like accounts in a banking system that records the cryptocurrency they hold. In blockchain systems, transactions are messages sending from one the initiator (source address) to the receiver (target address) [1].

The most well-known use of blockchain technology is on cryptocurrency platforms, such as Bitcoin and Ethereum. By preserving a secure and decentralized transaction record, its use on these cryptocurrency platforms ensures record authenticity, security, and confidence without needing a third party. Buterin, credited as the creator of Ethereum, was among the first to recognize the full potential of blockchain technology, which extended beyond enabling secure virtual payment methods. After Bitcoin, the Ethereum network's Ether (ETH) cryptocurrency is the second most popular digital currency [11].

Phishing is a well-known social engineering technique that tricks Internet users into disclosing private information that can be fraudulently used. Researchers have been working on detecting and preventing phishing on the Internet for the last two decades. Nevertheless, the primary environments have been emails [2] and websites [7], [8]. With the advancement of blockchain technology, phishing scams on cryptocurrency transactions have increased exponentially, necessitating a focus on detecting phishing in the virtual transaction environment.

Phishing detection methods in virtual transaction environments differ from traditional websites in target objects and data sources. Specifically, unlike on traditional websites, phishing detection focuses on distinguishing malicious web content, while on virtual transaction platforms, the focus is on detecting phishing addresses. In other words, while detecting phishing on traditional websites relies on the analysis of the content of the web page (URL, HTML, and network attributes), the detection framework in virtual transaction environments utilizes the transaction records between Ethereum addresses to distinguish between phishing and non-phishing addresses. Therefore, using phishing detection approaches for traditional phishing attacks on web pages and emails will be unsuitable for mitigating attacks on the Ethereum platform.

Existing phishing detection techniques on the Ethereum platform have focused on two approaches to detecting phishing addresses: 1. extracting statistical features from the amount and time stamp attributes, and 2. applying network embedding techniques to the above attributes. These approaches are based on the assumption that the amount of Ether sent between addresses and the record of time spent are the most important factors to consider when detecting phishing addresses. However, these approaches are limited as they depend on detecting large amounts of value because they imply a legitimate transaction. Using transaction amount as a criterion gives rise to a high misclassification of legitimate transactions with a low transaction amount. Also, phishing transactions in which significant amounts have been transacted are wrongly classified.

In this paper, we take a different approach to detecting phishing addresses on a virtual transaction platform. Intuitively, detecting phishing in the virtual transaction environment aims to alienate the bad actors. Therefore, instead of modelling the relationship between the transaction amount and the transaction time, we focused on the relationship between the addresses of the transactors to establish a pattern between them using statistical features. Our proposed approach is not dependent on the specific amount transacted but on the presence of any transaction to and from a suspicious node. Furthermore, the method proposed in this paper removes the extra complexity of using network embedding techniques while providing high AUC score.

Specifically, we propose a link prediction model that predicts whether a relationship exists between two transacting addresses on the Ethereum platform based on their node data. The node data in this paper comprises labelled node pairs (Ether transferring node address — Ether receiving node address) corresponding to possible transaction links and outputs 12 tailored features based on the node pairings. These features represent graph edges and are divided into positive and negative samples based on their node labels. Subsequently, the graph edges with corresponding labels are fed into a LightGBM classifier to obtain link predictions.

The main contributions of this work are as follows:

- This paper proposes a link prediction model which uses only the uses only the addresses of the receiving and sending addresses and extracts features from the local network on the Ethereum platform. The proposed approach is not dependent on the specific amount transacted but on the presence of any transaction to and from a suspicious node. Furthermore, the method proposed in this paper removes the extra complexity of using network embedding techniques while providing high recall and AUC score.
- The proposed framework's efficiency in identifying phishing nodes was validated by extensive experiments using real-world datasets from the Ethereum transaction network. Additionally, the results of the experiments demonstrate that the proposed link prediction method outperformed state-of-the-art feature-based node classification techniques.

The remainder of the paper is divided into the following sections: The next Section summarises related papers on proposed solutions for identifying phishing using traditional methods and elaborates on phishing identification on Ethereum. Section 3 discusses the proposed model in detail. Section 4 presents the research questions and evaluation criteria used to examine the proposed phishing detection framework. Section 5 contains the complete results of the proposed model's evaluations. Finally, Section 6 concludes the paper and discusses future work.

2 Related Works

Most state-of-the-art approaches to detect phishing transaction on Ethereum use graph embedding techniques. Graph modelling techniques have applied in many domains, the blockchain ecosystem is not left behind. Zhuang et al. [14] designed a graph classification algorithm to model semantic structures within smart contracts and detect inherent vulnerabilities within contracts. Liu et al. [5] proposed a GCN-based blockchain address classifier using graph analytics and an identity inference approach.

On the Ethereum platform, Wu et al. [12] proposed a technique for detecting fraud on the Ethereum platform using a concatenation of the statistical features extracted from the transaction amounts and timestamps and automated features from a novel network-embedding approach called trans2vec for downstream phishing classification.

Wang et al. [10] proposed a transaction subgraph network to identify ethereum phishing accounts (TSGN). The TSGN inspired by random walk uses a weight-mapping mechanism to retain the transaction amount information in the original transaction network for downstream network analysis tasks. 1621 transaction networks centered on phishing nodes and 1641 transaction networks centered on normal nodes were expanded into subgraphs using the proposed TSGN and applied to various graph classification algorithms, such as manual attributes, Graph2Vec, and Diffpool. Based on the deep learning method Diffpool, TSGN achieved the best classification performances of 94.35% and 93.64%, respectively.

Yuan et al. [13] approached phishing identification as a graph classification challenge, enhancing the Graph2Vec approach using line graphs and achieving a

high performance. The technique proposed by Yuan et al. focuses on structural elements extracted from line graphs, thereby omitting information from the graph direction, which is critical for identifying phishing schemes.

The study by Lin et al. [4]modeled the Ethereum network transaction data as a temporally weighted multi-digraph. These graphs were then applied to a random walk-based model to obtain explicable results regarding the interaction between network transactions for phishing account detection.

3 Methodology

This section elaborates on the architecture of the proposed phishing link prediction framework for Ethereum.

3.1 Problem Definition

Given a directed Multigraph, $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{Y})$ of a transaction network, where $\mathcal{V}$ represents a set of nodes that correlate to the target and source addresses on Ethereum. In this study, the source address is analogous to the address initiating the transaction, while the target address is the recipient. The variable $\mathcal{E}$ corresponds to the transaction relationship between the target and source addresses, where $\mathcal{E} = \{e_{u,w}, u, w \in V, u \neq w\}$. The edge attributes $e_{u,w}$ contains local network-based features $a_{u,w}$, such as node PageRanking, degree centrality, and betweenness centrality. $\mathcal{Y}_{A_x} \in \mathbb{R}^{|G| \times |\gamma|}$ is the label of each transaction in the Ethereum network. $\gamma = 1$ equates to a phishing transaction, while $\gamma = 0$ is a legitimate transaction on the Ethereum platform.

3.2 Proposed Phishing Detection Framework

Figure 1 provides an overview of the proposed phishing detection framework, which consists of three core parts: transaction graph construction and extraction of network features and the link phishing detection classifier.

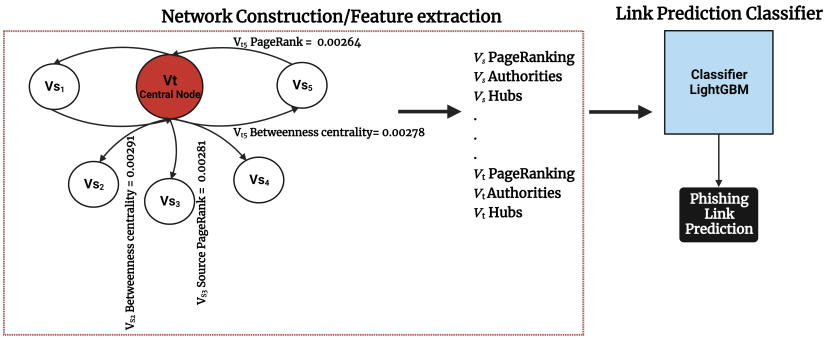

Fig. 1: The Phishing detection framework for transactions on the Ethereum platform.

Feature Extraction As shown in Figure 1, we first construct a large-scale Ethereum transaction network. The nodes are the network addresses, and the edges are the addresses' intrinsic local network-based characteristics. A transaction has two directions: out and in. The out-transactions of an account transfer ether from the account to other accounts and the in transactions of an account receive ether from other accounts. Specifically, the proposed model considers the relationship between the *From transaction address* (initiating node) and the *To transaction address* (receiving node) to determine the maliciousness of a transaction. With a target address V_t as a central node, we extract intrinsic features that relate the address V_t with all its corresponding source address V_s. Table 1 details the 12 features extracted.

Table 1: Description of features based on the local network

Features	**Description**
V_s PageRank	Ranking of the source nodes in the graph based on the number of transactions and the importance of the nodes making those transfers.
V_s Authorities	Estimates the source node value based on the incoming transactions.
V_s Hubs	Measures the source node value based on outgoing transactions.
V_s Betweenness centrality	Measures how often a source node V_s appears on the shortest paths between nodes in the network.
V_s Closeness centrality	Measures the average distance from a given source node to all other nodes in the network.
V_s Degree centrality	Measures the fraction of nodes V_s is connected to.
V_t PageRank	Ranking of the target nodes in the graph based on the number of transactions and the importance of the nodes making those transfers.
V_t Authorities	Estimates the value of V_t based on the incoming transactions.
V_t Hubs	Measures the value of V_t based on outgoing transactions.
V_t Betweenness centrality	Measures how often a target node V_t appears on the shortest paths between neighbouring nodes in the network.
V_t Closeness centrality	Measures the average distance from a given target node to all other nodes in the network.
V_t Degree centrality	Measures the fraction of nodes V_t is connected to.

The PageRank feature, obtained from both the source and target nodes, ranks the given nodes according to the number incoming relationships and the importance of the corresponding source nodes. PageRank is essential because it rates each node based on its in-degree (the number of transactions transferred to the node) and its out-degree (the number of transactions transferred by the specified node).

The HITS algorithm is one of the essential link analysis algorithms. It produces two primary outcomes: authorities and hubs. In this study, the HITS algorithm calculates the worth of a node by comparing the number of transactions it receives (authorities) and the number of transactions it originates (hubs). As the primary objective of phishing addresses is to obtain as much ETH as possible, and they may not transmit any ETH, the value of the Authorities and Hubs will play a crucial part in distinguishing phishing addresses from legitimate ones.

Degree centrality offers a relevance score to each node based on the number of direct, 'one hop' connections it has to other nodes. In an Ethereum network, we assume that legitimate nodes are more likely to have faster connections with nearby nodes and a higher degree of centrality value. This assumption is based

on the observation that there are likely more legitimate nodes in a given network that phishing nodes.

$$d_v = \frac{deg(v)}{n}, \text{for} v \in V \tag{1}$$

Where 'deg(v)' is the degree of node 'v' and 'n' is the number of nodes in set V.

In an Ethereum network, **betweenness centrality** quantifies the frequency with which a node is located on the shortest path between other nodes. Specifically, this metric identifies which nodes are "bridges" connecting other nodes in a network. This is achieved by placing all the shortest pathways and then counting how often each node falls on one. Phishing nodes are more likely to have a low betweenness centrality rating because they may have less of an impact on the network.

$$c_B(v) = \sum_{p,q \in V} \frac{\sigma(p,q|v)}{\sigma(p,q)} \tag{2}$$

where V is the set of nodes, $\sigma(p,q)$ is the number of shortest (p,q)-paths, and $\sigma(p,q|v)$ is the number of those paths passing through some node v other than p,q.

Essentially, **closeness centrality** assigns a score to each node based on its "closeness" to every other node in the network. This metric computes the shortest pathways connecting all nodes and provides each node with a score based on the sum of its shortest paths. The analysis of proximity centrality values revealed that the network nodes are more likely to affect other nodes rapidly.

$$C(u) = \frac{n-1}{\sum_{v=1}^{n-1} d(v,u)} \tag{3}$$

where 'd(v, s)' is the shortest-path distance between 'v' and 's', and 'n' is the number of nodes in the graph.

The Link Prediction Classifier As stated earlier, the objective of link prediction is to determine the presence of phishing transaction using the intrinsic features from the local network. Subsequently, we employed the the LightGBM classifier for the downstream task to detect phishing transactions. Please note that other shallow machine learning classifier can be used at this stage. However, we chose to use LightGBM because research has shown that it provides a faster training speed and more efficiency when compared to other shallow machine learning algorithms [3]. In addition, it utilizes less memory and has a higher degree of precision than all other boosting techniques. They have also been proven to be compatible with larger datasets [6].

4 Research questions and Experimental Setup

This section discusses the research questions, dataset, hyperparameters and metrics used to set up and evaluate the proposed model and its baselines.

4.1 Research questions

- **RQ1:** How accurate is the proposed link prediction model for detecting phishing transactions compared with other time and amount feature-based state-of-the-art approaches?
- **RQ2:** What are the technical alternatives to the proposed link prediction model, and how effective are they?
- **RQ3:** How important are the features used in the proposed link prediction model for detecting phishing transactions between Ethereum addresses?

Data Source/ Preprocessing The dataset used in this paper was obtained from the xblock.pro website[4]. It contains 1,262 addresses labelled phishing nodes, and 1,262 non-phishing nodes crawled from Etherscan. Each address contains the transaction information between the target node and its corresponding source nodes. Note that transactions exist between a specific target node and multiple source nodes. This observation is not surprising because a single phishing address can receive multiple ETHs from different non-phishing addresses.

Existing studies use only the first node address and ether received for graph construction. This study aims to investigate beyond the first-node address and explore all transaction records carried from and to the addresses. This approach removes the challenges of a few datasets and demonstrates the importance of studying the connectivity between outgoing and incoming transactions from phishing and non-phishing nodes.

Consequently, 13,146 transactions were extracted from 1262 phishing addresses and 286,598 transactions from 1262 legitimate addresses. As it is clear that the number of legitimate transactions is considerably higher than the number of phishing transactions, the synthetic minority oversampling technique (SMOTE) was adopted to address the imbalance in the training set. Specifically, a random number of minority classes was added until both classes were equally represented. To prevent bias in the results, the instances were normalized in the dataset to appear similar across all records, leading to cohesion and higher data quality.

In total, after oversampling the minority class, our final corpus contained a balanced dataset of **286,598** phishing and benign instances.

Hyperparameter Setting A combination of hyperparameters is required to classify the link prediction model using LightGBM. A grid search was used to determine the optimal hyperparameters of the models by setting the number of estimators to 10,000 and the learning rate to 0.02. In addition, the default value for the number of leaves was set at 31 and set the application type to binary.

Evaluation Metrics The performance of the link prediction model was evaluated using $Recall = \frac{(TP)}{(TP+FN)}$ and $F1_{score} = \frac{(Precision \times Recall)}{(Precision+Recall)}$ where TP, FP and FN represent the numbers of True Positives, False Positives and False Negatives, respectively. Also, the Area Under the Curve (AUC) score was calculated, representing the degree or measure of separability. A model with a higher AUC is better at predicting True Positives and True Negatives. Finally, to assess the performance of the proposed model and its baseline on the corpus, the dataset was divided into 80% for training and 20% for testing.

[4] http://xblock.pro/#/dataset/6

5 Results

This section discusses the experiments conducted to evaluate the proposed phishing link prediction method and the results of answering each research question.

5.1 Comparing The Proposed Model with State-Of-The-Art Baselines (RQ1)

To demonstrate a thorough evaluation of our methods, a comparison of the performance of the link prediction model with the existing state-of-the-art feature-based approaches was conducted. These methods include those utilized by Wu et al. [12], who used non-embedding techniques to extract local information from addresses to detect phishing. The time features, amount features, and time plus amount feature are among the retrieved features.

Table 2: Result of the proposed model and other state-of-the-art non-embedding models

Models	Recall	F-1 Score	AUC Score
Proposed Model	**0.890**	**0.697**	**0.930**
[12](Time Features Only)	0.302	0.326	0.835
[12](Amount Features Only)	0.321	0.358	0.848
[12](Time + Amount Features)	0.478	0.494	0.865

Table 3: Result of the proposed model and its alternative options

Models	Recall	F-1 Score	AUC Score
Proposed Model	**0.890**	**0.697**	**0.930**
Logistics Regression	0.838	0.146	0.694
Naive Bayes	0.982	0.106	0.605
Decision Tree	0.865	0.467	0.890

Result: Table 2 presents the outcomes of the approaches (balancing recall, F1-Score and AUC score). The proposed model demonstrated the best recall performance for this dataset. The results indicate that the proposed method can detect phishing transactions with a satisfactory level of recall and AUC score by utilizing only locally based information collected from analysis of the relationship between the transaction addresses.

The proposed model also performed the best in the F1-score, demonstrating that the phishing class's overall precision and recall performance is robust. In other words, the proposed model not only detects phishing cases accurately but also avoids incorrectly labelling too many legitimate addresses as phishing. This shows that the proposed strategy for phishing strikes a balance between precision and recall. Compared to the other models, the time-features-only model performed the worst, indicating that it could not correctly identify most of the phishing class.

Investigating False Positives and False Negatives From the results in Section 5.1, we found that the proposed model inaccurately classified 287 legitimate links as phishing links and 2702 phishing instances were incorrectly classified as legitimate. To investigate false positive links (i.e., legitimate transactions that were wrongly classified as phishing) and false negatives (i.e., phishing transactions that were incorrectly identified as legitimate), we performed a manual analysis on a subset of 100 addresses and their corresponding edges from the false positives and false negatives obtained from the result discussed above.

Our analysis shows that most false-positive and false-negative transactions involve phishing addresses transferring ETH to a legitimate address. This type of transaction is uncommon and only occurs when the phishing address attempts to establish credibility with the legitimate target address. Although this type of transaction is genuine, as the legitimate address duly receives the ETH, the model is bound to misclassify it because it originates from a phishing address. Exploring the maliciousness of specific addresses in the Ethereum network and determining their validity will be a top priority for future work.

5.2 Alternative Technical Options for The Proposed Link Prediction Model (RQ2)

The selected shallow machine learning classifier of the detection framework also influences the detection performance. Consequently, this study considers logistic regression, naive Bayes, and decision trees as the baseline classifiers. Using the extracted features as input, Table 3 details the detection outcomes of the three classifiers using the extracted features as input.

Result: From the results, it is clear that the performance of the predicted model using the LightGBM classifier is superior to that of other classifiers owing to its suitability for the link prediction task. The proposed model produced an average F-1 score, a recall rate, and an AUC score of 83%. Across all the evaluative parameters, logistic regression was the alternative option with the lowest performance. This low performance is because logistic regression requires modest or no multicollinearity among independent variables.

5.3 Feature Importance (RQ3)

In addition to our analysis, an investigation of the features that were informative for the classification outcomes of the proposed model was conducted. We employed a sensitivity analysis technique to determine the impact of each feature on categorization output. In sensitivity analysis, the variability of changes in results is determined by the input variability [9]. In this study, the effect of each feature was determined using a one-at-a-time method. This strategy measures the model output statistics for each entry change category. The efficiency of each feature is then estimated based on the sensitivity of the classification model.

Result: In Table 4, it is evident that the absence of the target node's closeness centrality and target degree had the most significant impact on the model's declining recall. Eliminating the source node betweenness centrality and target PageRank had the opposite effect on the link prediction model's recall. With the

Table 4: Result of The Sensitivity Analysis

Features	Recall	F-1 Score
V_s PageRank	0.885	0.696
V_s Authorities	0.884	0.698
V_s Hubs	0.885	0.693
V_s Betweenness centrality	0.887	0.696
V_s Closeness centrality	0.884	0.695
V_s Degree centrality	0.884	0.694
V_t PageRank	0.886	0.688
V_t Authorities	0.883	0.698
V_t Hubs	0.883	0.695
V_t Betweenness centrality	0.883	0.688
V_t Closeness centrality	0.881	0.694
V_t Degree centrality	0.883	0.697

removal of the source hub, the model's F1-Score and recall of the model are unaffected. Not analyzing the target node's PageRank and betweenness centrality reduced the F1-score by approximately 0.008. Therefore, removing these features reduced the effectiveness of the model.

In summary, the most significant characteristics of the proposed model are the target node's (recipients) PageRank, betweenness, and closeness centrality's. Eliminating these three features reduces the F1-score by approximately 0.008. This result is not surprising given that the primary objective of attackers on the Ethereum platform is to coerce victims to send them ETH.

5.4 Limitations

This study has some limitations. First, the proposed feature sets depend entirely on a specific dataset and may not be easily adapted to another dataset without minor adjustments. Second, network embedding techniques, such as Node2Vec and trans2Vec, might automate the feature extraction process from large-scale network data.

Nevertheless, network-embedding models have a higher resource consumption. In addition, unlike our proposed model, which uses features extracted from the local network, network embedding techniques are challenging to explain. Also, timestamps can easily be added to evolve the phishing detection technique into a time-series classification.

6 Conclusion and Future Work

This paper proposes a systematic study for detecting phishing transactions in an Ethereum network using link prediction. Specifically, a three-step approach for identifying the connections between network nodes using extracted local network features were demonstrated. We extracted 12 features based on the influence and connections between the addresses in the network and used them as inputs for a LightGBM classifier. Experiments on real-world datasets demonstrated the effectiveness of the proposed link prediction model over existing feature-based

state-of-the-art models in detecting phishing transactions. In future, we intend to conduct further studies on the impact of the proposed link prediction model on other downstream tasks such as gambling, money laundering, and pyramid schemes.

References

1. Chen, W., Guo, X., Chen, Z., Zheng, Z., Lu, Y.: Phishing scam detection on ethereum: Towards financial security for blockchain ecosystem. In: IJCAI. pp. 4506–4512. ACM (2020)
2. Gutierrez, C.N., Kim, T., Della Corte, R., Avery, J., Goldwasser, D., Cinque, M., Bagchi, S.: Learning from the ones that got away: Detecting new forms of phishing attacks. IEEE Transactions on Dependable and Secure Computing **15**(6), 988–1001 (2018)
3. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.Y.: Lightgbm: A highly efficient gradient boosting decision tree. Advances in neural information processing systems **30** (2017)
4. Lin, D., Wu, J., Xuan, Q., Chi, K.T.: Ethereum transaction tracking: Inferring evolution of transaction networks via link prediction. Physica A: Statistical Mechanics and its Applications **600**, 127504 (2022)
5. Liu, X., Tang, Z., Li, P., Guo, S., Fan, X., Zhang, J.: A graph learning based approach for identity inference in dapp platform blockchain. IEEE Transactions on Emerging Topics in Computing (2020)
6. Minastireanu, E.A., Mesnita, G.: Light gbm machine learning algorithm to online click fraud detection. J. Inform. Assur. Cybersecur **2019** (2019)
7. Opara, C., Chen, Y., et al.: Look before you leap: Detecting phishing web pages by exploiting raw url and html characteristics. arXiv preprint arXiv:2011.04412 (2020)
8. Opara, C., Wei, B., Chen, Y.: Htmlphish: Enabling phishing web page detection by applying deep learning techniques on html analysis. In: 2020 International Joint Conference on Neural Networks (IJCNN). pp. 1–8. IEEE (2020)
9. Pannell, D.J.: Sensitivity analysis of normative economic models: theoretical framework and practical strategies. Agricultural economics **16**(2), 139–152 (1997)
10. Wang, J., Chen, P., Yu, S., Xuan, Q.: Tsgn: Transaction subgraph networks for identifying ethereum phishing accounts. In: International Conference on Blockchain and Trustworthy Systems. pp. 187–200. Springer (2021)
11. Wood, G., et al.: Ethereum: A secure decentralised generalised transaction ledger. Ethereum project yellow paper **151**(2014), 1–32 (2014)
12. Wu, J., Yuan, Q., Lin, D., You, W., Chen, W., Chen, C., Zheng, Z.: Who are the phishers? phishing scam detection on ethereum via network embedding. IEEE Transactions on Systems, Man, and Cybernetics: Systems (2020)
13. Yuan, Z., Yuan, Q., Wu, J.: Phishing detection on ethereum via learning representation of transaction subgraphs. In: International Conference on Blockchain and Trustworthy Systems. pp. 178–191. Springer (2020)
14. Zhuang, Y., Liu, Z., Qian, P., Liu, Q., Wang, X., He, Q.: Smart contract vulnerability detection using graph neural network. In: IJCAI. pp. 3283–3290 (2020)