Blockchain Scam Detection: State-of-the-art, Challenges, and Future Directions

Shunhui Ji[1], Congxiong Huang[1], Hanting Chu[1]
Xiao Wang[1], Hai Dong[2], and Pengcheng Zhang[1]✉

[1] College of Computer and Information, Hohai University, Nanjing, China
[2] School of Computing Technologies, RMIT University, Melbourne, Australia

Abstract. With the rapid development of blockchain platforms, such as Ethereum and Hyperledger Fabric, blockchain technology has been widely applied in various domains. However, various scams exist in the cryptocurrency transactions on the blockchain platforms, which has seriously obstructed the development of blockchain. Therefore, many researchers have studied the detection methods for blockchain scams. On the basis of introducing the mainstream types of scams, including Ponzi scheme, Phishing scam, Honeypot, and Pump and dump, this paper provides a thorough survey on the detection methods for these scams, in which 48 studies are investigated. The detection methods are categorized into the analysis-based methods and the machine learning-based methods in terms of the adopted techniques, and are summarized from multiple aspects, including the type of dataset, the extracted feature, the constructed model, etc. Finally, this paper discusses the challenges and potential future research directions in blockchain scam detection.

Keywords: Blockchain; Smart contract; Ponzi scheme; Phishing scam; Honeypot; Pump and dump; Scam detection.

1 Introduction

Blockchain [1], as a distributed ledger technology, demonstrates the advantages of data immutability, security, transparency, and anonymity, which has facilitated the development of various applications. Blockchain-based trading platforms allow users to trade cryptocurrencies without being supervised by a third party. Smart contracts have extended the functionality of blockchain. Users deploy customised smart contracts on the blockchain platform and the smart contracts will be executed automatically according to their embedded programs.

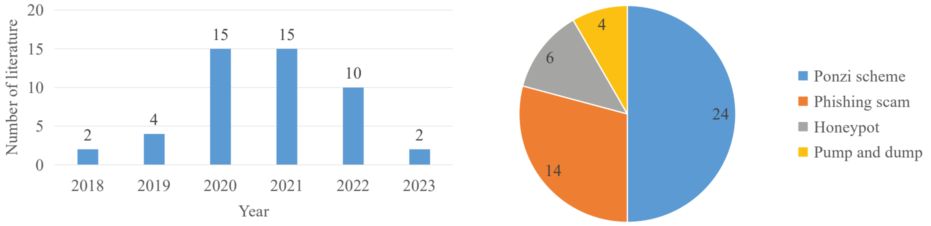
However, various financial scams were launched on blockchain platforms to defraud users, which may compromise users' account passwords and transfer their assets for illicit profits. According to the survey [2], cryptocurrency crime with illicit addresses on the blockchain platform earned $7.8 billion in 2020, while the value reached $14 billion in 2021. The increasing scams on blockchain platforms have caused huge financial losses and a crisis of trust in the blockchain

Ji, S., Huang, C., Chu, H., Wang, X., Dong, H., Zhang, P. (2024). Blockchain Scam Detection: State-of-the-Art, Challenges, and Future Directions. In: Chen, J., Wen, B., Chen, T. (eds) Blockchain and Trustworthy Systems. BlockSys 2023. Communications in Computer and Information Science, vol 1896. Springer, Singapore. https://doi.org/10.1007/978-981-99-8101-4_1

environment, which has seriously obstructed the development of blockchain. Therefore, many researchers have conducted research on scams on the blockchain and proposed corresponding detection methods. Some detection methods aim at specific types of scams, such as Ponzi scheme [3] and Honeypot [4]. And some detection methods do not target at specific types, detecting abnormal transactions between blockchain addresses and consequently identifying malicious and illegal entities [5] that may be involved in scams, gambling, etc.

Specifically, Li et al. [6] surveyed works about the blockchain anomaly detection using data mining technology. Blockchain anomaly detection methods were analyzed and sorted from two aspects: general detection methods and specific detection methods. And some of the detection methods for Ponzi schemes and Pump and dumps were summarised in specific detection methods. Bartoletti et al. [7] reviewed studies on cryptocurrency scams, in which cryptocurrency scams were classified and summarized. Aida et al. [8] surveyed studies on anomaly and fraud detection using data mining techniques with blockchain big data, in which the key elements of these methods were summarized and analyzed. Wu et al. [9] reviewed studies on network-based anomaly detection methods for cryptocurrency transaction networks, in which the works were summarized from four perspectives: entity identification, transaction pattern identification, illegal activity identification, and transaction tracking.

The aforementioned works mainly summarized the works from the perspective of *anomaly detection*, with less focus on specific *scam detection*. The scope of anomalies is broad and its boundary is not clearly defined. In contrast, scam has a clear definition and limited scope, which can be considered as a type of anomaly. In this regard, this paper provides a systematic review and analysis for the studies on the detection of specific blockchain scams, including Ponzi scheme, Phishing scam, Honeypot, and Pump and dump.



(a)

(b)

Fig. 1. Distribution of papers in terms of publication time and scam type

To systematically investigate the related studies, firstly, the keywords ”Ponzi scheme”, ”Blockchain scam detection”, ”Phishing scam”, ”Honeypot”, ”Pump and dump”, etc. were set to search for relevant papers on major academic search engines (e.g. Google Scholar, DBLP, Web of science). Secondly, we screened these papers and retained papers that were highly relevant to the research question and of good quality. Thirdly, the related works and references in these papers were reviewed to search for additional relevant papers. Finally, 48 papers were selected, including Ponzi scheme detection (24 papers), phishing scam detection (14 papers), Honeypot detection (6 papers), and Pump and dump detection (4 papers). The distribution of papers in terms of publication time and scam type are respectively shown in Figure 1 (a) and (b).

The rest of this paper is organized as follows. Section 2 introduces the mainstream types of scams on the blockchain. Sections 3 and 4 provide detailed overviews of the analysis-based detection methods and the machine learning-based detection methods, respectively. Section 5 discusses the inadequacy of existing scam detection methods and the further research directions. Finally, section 6 concludes the paper.

2 The mainstream types of scams

2.1 Ponzi scheme

Ponzi scheme is a fraudulent investment trap, also known as a high yield investment project. Ponzi scheme usually promises investors a return on investment that is much higher than that of ordinary projects, thus attracting a large number of investors to participate in the project. However, the profit return mechanism of the scheme is using the investment funds of later investors as the return for previous investors, with which the creator of the scheme and previous investors usually make a huge profit. For the other investors, it needs more investors to participate for obtaining the promised benefits, which becomes increasingly difficult to achieve. When the number of investors reaches the bottleneck, the scam will collapse and the later participants will lose their investment [3].

2.2 Phishing scam

Phishing scam occurs a lot in the transactions on the Internet. Scammers use various means, such as creating fake websites that look legitimate, sending emails from trusted sources, to induce users to open URLs or attachments that contain malicious software. Then malicious software will be installed to monitor user behaviour for stealing private information, such as bank card number and password, with which the money will be withdrawn from the user's account. Scammers also may disguise themselves as trustworthy organizations, enticing users to transfer money to their accounts. The widespread popularity of cryptocurrencies and the convenience of trading on blockchain platforms have made blockchain a new target for phishing scams. Phishing scams on blockchain spread through fake emails, websites, and decentralised applications, stealing users' private keys and transferring assets to the designated address [10].

2.3 Honeypot

Honeypot is a new type of scam on Ethereum, which utilizes the difficulties of smart contract programming technology to create a series of traps to confuse users who are not familiar with the mechanism of Solidity and the EVM (Ethereum Virtual Machine). Honeypot makes the users mistakenly believe that they understand the logic of the smart contract code and can benefit from sending cryptocurrency to the smart contract, but ultimately they will lose all their investment.

The scammer designs a smart contract with obvious vulnerabilities or intuitive conditions for obtaining benefits. It seems that exploiting existing vulnerabilities or conditions will definitely result in benefits. Then the source code of the smart contract is published on the blockchain platform and advertised widely to attract users who have a certain understanding and programming ability in smart contracts and try to profit from them. These users are attracted by the apparent vulnerability and the profit conditions, but do not notice the deeper trap hidden behind the code. Once the user invests cryptocurrency into the smart contract, the contract will not work as expected and the invested cryptocurrency cannot be withdrawn. Eventually, the creator of the smart contract extracts all cryptocurrencies through a designed backdoor [4].

2.4 Pump and dump

Pump and dump is a price manipulation scam. In the cryptocurrency trading market, scammers purchase the cryptocurrencies at a lower or average price for a period of time. And then they artificially raise the price of the cryptocurrency and entice other investors to buy at a higher price by spreading false information on social media. After the scammers sell their holdings of cryptocurrency, the price usually falls, causing significant losses for buyers. This kind of scam typically targets micro and small cryptocurrencies because they are more easily manipulated [11].

3 Analysis-based detection methods

For the smart contracts which are successfully deployed on the Ethereum platform, the bytecodes, ABI (Application Binary Interface), transaction records, and other information are available on the Etherscan website[3], and the verified Solidity source codes of some smart contracts are released[4]. With the source code of smart contract, program analysis techniques can be used to detect scams. With the transaction records, transaction analysis can be performed. The existing analysis-based detection methods mainly aim at two types of scams: Ponzi scheme and Honeypot.

3.1 Ponzi scheme detection

Chen et al. [12] proposed a semantic-aware detection method SADPonzi based on a heuristic guided symbolic execution technique. The execution paths and symbolic contexts for the investment and return behaviours in a contract were generated and combined with internal call relations to obtain overall semantic information. The semantic information was then combined with the control flow graph of the opcode to analyze whether Ponzi scheme exists through the mode of fund allocation.

[3] https://etherscan.io/

[4] https://docs.soliditylang.org/en/v0.8.19/

Bartoletti et al. [3] proposed a Ponzi scheme detection method based on string similarity analysis. NLD (Normalised Levenshtein Distance) algorithm was used to calculate the similarity of the bytecodes between the target smart contract and Ponzi scheme contract. Contracts whose NLD value is less than 3.5 were considered as potential Ponzi schemes, which filtered out the simple and repetitive scam contracts.

Sun et al. [13] proposed a detection method PonziDetector based on similarity analysis for contract behaviour forest. Software testing technique was used to uncover the contract behaviour during the interactions between the test cases and the smart contract, which was then described as the behavioural forest. Then the similarity between the behavioural forests was calculated by the AP-TED (All Path Tree Edit Distance) algorithm to detect Ponzi scheme by setting the similarity threshold.

Song et al. [14] detected Ponzi scheme by analyzing the amount of funds transferred between various accounts. A virtual transaction network was created to simulate trading behaviour among accounts. They divided the accounts in the transaction network into different types of accounts, such as master account, investing account, partner account, etc. Then a series of rules were set to compare the amount of Ether transferred between different accounts to determine whether a contract is Ponzi scheme.

3.2 Honeypot detection

HoneyBadger [4, 15] is a tool that employs symbolic execution and heuristics to detect Honeypots. With bytecode as input, it constructs the control flow graph and symbolically executes its different paths to perform symbolic analysis. With the results of the symbolic analysis, cash flow is analyzed to detect whether the contract is capable to receive as well as transfer funds. Finally, different honeypots are identified via heuristics.

Ji et al. [16] proposed a Honeypot detection method based on anisotropic features. They refined the attack model of honeypots and summarized the complete attack process. Then they conducted feature mining on ten honeypot families to extract the anisotropic features and constructed the honeypot genealogy. With the guidance of honeypot genealogy, the anisotropic feature matching was performed to detect Honeypots.

4 Machine learning-based detection methods

The key of machine learning-based detection methods is feature extraction from the smart contracts or the transaction records in the dataset. For the smart contract codes, feature extraction is usually carried out using language processing techniques. Some studies treat the scam detection problem as image classification problem by transforming bytecodes into images. For the transaction records, transaction features are mainly extracted by analyzing the trading behaviour among accounts, such as fund allocation, account balance changing, etc. Transaction network model construction can also be used to extract feature information of nodes and edges in the network model.

4.1 Ponzi scheme detection

Table 1 shows the existing machine learning-based detection methods for Ponzi scheme, which lists the types of extracted features and the classification models used for detection. According to the type of the dataset, machine learning-based detection methods for Ponzi scheme are classified into five categories: opcode, bytecode, source code, transaction record, and combined data oriented detection.

Table 1. Summary of machine learning-based detection methods for Ponzi scheme.

Data	Ref.	Feature	Model
Opcode	[17]	n-gram TF	PonziTect
	[18]	n-gram TF	OB
	[19]	n-gram TF,TF-IDF, TF-OUR	LR, DT, SVM, ET,etc.
Bytecode	[20]	image	CNN
	[21]	matrice	OC-SVM, IF
Source code	[22]	AST	DT, SVM
	[23]	token sequence	Transformer
Transaction record	[24]	Tran-Feature	PIPPER, BN, RF
	[25]	Tran-Feature	GCN
Combined data	[26]	TF, Tran-Feature	XGBoost
	[27]	TF, Tran-Feature	DT
	[28]	TF, Tran-Feature	J48, RF
	[29]	TF, Tran-Feature	LSTM
	[30]	TF, Tran-Feature, bytecode similarity	LightGBM
	[31]	TF, Tran-Feature, sequence	CTRF
	[32]	TF, sequence, ABI call	SE-CapsNet
	[33]	2-gram TF-IDF, bytecode similarity	CatBoost
	[34]	TF, n-gram TF-IDF, sequence, counting vector	BT, XGBoost, etc.
	[35]	Tran-Feature, node feature	GCN
	[36]	numerical feature	XGBoost

Opcode oriented detection. Opcode, which is obtained by disassembling bytecode, is the most common type of data used in Ponzi scheme detection. In the existing works, the features extracted from opcode include n-gram TF, TF-IDF (Term Frequency - Inverse Document Frequency), and TF-OUR.

The n-gram TF feature is the frequency of the term which is combined by n consecutive opcodes. Fan et al. [17,18] extracted n-gram TF features ($1 \leq n \leq 4$) after eliminating stop words and splitting opcodes. With the extracted features, the PonziTect algorithm [17] and the OB (Ordered Boosting) algorithm [18] were used to detect Ponzi scheme contracts.

TF-IDF measures the importance of each opcode for a document within a set of documents. The more frequently an opcode appears in a document, but the less frequently it appears in other documents, the more representative it is for that document. Peng et al. [19] proposed TF-OUR feature, which was calculated by dividing the frequency of the term by the frequency of the maximum prefix term for the term. Eight classification models, such as DT (Decision Tree), RF (Random Forest), LR (Logistic Regression), etc., were combined with n-gram TF, TF- IDF, and TF-OUR features ($1 \leq n \leq 5$) to explore the best solution for Ponzi scheme detection.

Bytecode oriented detection. The bytecode of smart contract is compiled from the source code which is programmed with the high-level language, such as Solidity. It consists of a series of hexadecimal numbers that are highly sequential and abstract.

Lou et al. [20] used code visualisation technique to convert bytecodes into images, which were then employed to construct CNN (Convolutional Neural Network) model for Ponzi scheme detection, in which spatial pyramidal pooling approach was used to make the CNN adapt to different sizes of images.

Shen et al. [21] transformed the bytecodes into high-dimensional matrices by taking two bytes as one feature value. The PCA (Principal Component Analysis) algorithm was then used to reduce the dimensionality of the matrices and eliminate the noise in the matrices. Finally, the OC-SVM (One-class-Support Vector Machine) and IF (Isolation Forest) models were constructed to detect Ponzi schemes.

Source code oriented detection. Compared with opcodes and bytecodes, the source codes of smart contracts have more intuitive semantic information, which can be transformed into AST (Abstract Syntax Trees) for further processing.

Ibba et al. [22] parsed the structure of the AST and replaced all mathematical operators with corresponding semantic meaning. With the AST transformed into semantic document, the DT and SVM (Support Vector Machine) models were constructed to perform classification.

Chen et al. [23] converted the AST to specially formatted code token sequences. The Transformer model was employed to learn the structure features, semantic features, and long-range dependencies in the code token sequences for classification.

Transaction record oriented detection. Transaction records of blockchain addresses also can offer related features to distinguish between Ponzi and non-Ponzi schemes.

Bartoletti et al. [24] extracted Tran-Features (Transaction features), including the Gini coefficient of the values transferred to the address, the lifetime of the address, etc, to detect Ponzi schemes on Bitcoin with three classification models: RIPPER, BN (Bayes Network) and RF.

Yu et al. [25] collected the transaction records to establish TN (Transaction Network), from which Tran-Features were extracted. And the GCN (Graph Convolutional Network) model was constructed for Ponzi scheme detection.

Combined data oriented detection. In the existing studies about Ponzi scheme detection, many researchers extracted multiple features from various data to construct detection models.

Chen et al. [26–29] extracted opcode TF and Tran-Feature features to construct the Ponzi scheme detection models. Except for the TF and Tran-Feature features, the similarity of the bytecodes between the target smart contract and Ponzi scheme contract was incorporated to construct the LightGBM model [30], and the sequence of bytecode was incorporated to construct the CTRF (Code and Transaction Random Forest) model [31].

In addition, in the detection model construction, the TF, sequence of bytecode, and ABI call features were extracted [32]; the 2-gram TF-IDF and bytecode similarity features were extracted [33]; the features, including TF, n-gram TF-IDF ($2 \leq n \leq 3$), sequence and count vector which represents the occurrence of each word in the bytecode, were extracted [34]; the Tran-Feature features from the transaction records and the node features from auxiliary heterogeneous interaction graph, which contains the information of externally owned account and contract account, were extracted [35].

Fan et.al [36] extracted numerical features associated with DApp (Decentralized Application) submitter's information, such as transaction time and investor's address, to construct the detection model. To protect the user's sensitive information, the clients of different DApps jointly trained the XGBoost model by federal learning without sharing the original data.

4.2 Phishing scam detection

The machine learning-based detection methods for Phishing scam usually extract features from the transaction records to construct the detection model. Table 2 provides a summary of the machine learning-based detection methods for phishing scam, listing the representation network of transactions, the feature extraction methods, the types of extracted features, and the classification models.

Table 2. Summary of machine learning-based detection methods for Phishing scam.

Ref.	Network	Extraction method	Feature	Model
[37]	TN	node2vec	latent	OC-SVM
[38]	TN	Line_graph2vec	latent	SVM
[39]	TN	trans2vec	node	SVM
[40]	TN	INSS	node	GCN
[41]	TN	MP	node	GCN
[42]	TN	LTFE	network,time series	LR
[43]	TN	manual-designed	account,network	SVM,KNN,AdaBoost
[10]	TN	GCF	cascade	DElightGBM
[44]	TN	GCN	structural	LightGBM
[45]	TN/TSGN/ Directed-TSGN	Graph2Vec, Diffpool	node/ handcrafted	RF
[46]	Ego-Graph	node relabeling	structural,attributed	DT
[47]	TTAGN	edge2node	structural	LightGBM
[48]	SRG	GCN	edge	LR,SVM,XGBoost
[49]	TPG	GNN	transaction pattern	MCGC

Many researchers convert transaction records into TN (Transaction Network) or TG (Transaction Graph), which are referred as TN in this paper, to extract features using graph embedding or self-defined algorithms. Yuan et al. [37, 38] respectively used the node2vec algorithm and the Line_graph2vec algorithm to extract the latent features to construct the OC-SVM and SVM models. The trans2vec algorithm [39] which takes the transaction amount and timestamp into consideration, the INSS (important neighbors subgraph sampling) method [40] which extracts features from the neighbor information of node, and the MP (Message Passing) algorithm [41] which computes the passed information between the nodes, were used to extract the node features. Wan et al. [42] used LTFE (Local network structures and Time series of transactions feature extraction) method to extract network features and time series features. Wen et al. [43] extracted the account features (such as the number of large transactions) and network features (such as the number of in-degree neighbors) to construct the SVM, KNN (K-Nearest Neighbor) and AdaBoost models. Chen et al. [10] used the GCF (Graph-based Cascade Feature extraction) method to extract cascade features, including node features and n-order features which contain the node features of the n-order friends, to construct DElightGBM (lightGBM-based Dual-sampling Ensemble algorithm) model. Chen et al. [44] used GCN (Graph Convolutional Network) to extract the structural features for constructing the LightGBM model.

Wang et al. [45] extracted TSGN (Transaction SubGraph Network) from TN, and expanded TSGN to Directed-TSGN which introduces the direction attributes. Then they respectively used graph2vec and Diffpool algorithm to extract node features from TN, TSGN, and Directed-TSGN. They also extracted handcrafted features from the three networks. The RF models constructed with the nine sets of features were compared. Xia et al. [46] constructed k-hop directed Ego-Graph for each address. Then they used a graph embedding method based on node relabeling strategy to extract both structural and attributed features. Li et al. [47] proposed TTAGN (Transaction Aggregation Graph Network), in which the edge representations around the node to fuse topological interactive relationships were aggregated into the representation of the node. Then they used edge2node algorithm to extract structural features. Fu et al. [48] transformed the TG into SRG (Sender and Receiver Graph), and used GCN to learn edge features in the graph. Zhang et al. [49] extracted transaction pattern features with GNN (Graph Neural Network) from TPG (Transaction Pattern Graph), which reduced the computational complexity through graph classification, to construct the MCGC (Multi-Channel Graph Classification) model.

4.3 Detection of other scams

Table 3 shows a summary of machine learning-based detection methods for other scams, including Honeypot and Pump and dump, listing the types of extracted features and the classification models.

Honeypot. The existing honeypot detection methods are all based on code features or transaction features to construct the detection model. Chen et al. [50] extracted n-gram TF features ($1 \leq n \leq 3$) from the opcode to construct the LightGBM model. Hara et al. [51] employed to extracted the distributed representation features from bytecode with the word2vec method and the TF-IDF features from opcode to construct the XGBoost model. Camino et al. [52] extracted the source code features (such as the number of lines in the source code) and the transaction features to construct the XGBoost model.

Table 3. Summary of machine learning-based detection methods for other scams.

Scams	Reference	Feature	Model
Honeypot	[50]	n-gram TF	LightGBM
	[51]	TF-IDF, distributed representation	XGBoost
	[52]	source code, Tran-Feature	XGBoost
Pump and dump	[11]	coin feature	RF, GLM
	[53]	moving window	RF, LR
	[54]	moving window	RF, AdaBoost
	[55]	social	CNN, LSTM

Pump and dump. Xu et al. [11] extracted coin features before Pump and dump in the dataset based on the analysis of the market movements of coins. Then the coin features were employed to construct the RF, GLM (Generalized Linear Model) models for Pump and dump detection. Morgia et al. [53,54] split the historical trading data in chunks of seconds and defined a moving window. The moving window related features, such as the moving standard deviation of the number of trades, were extracted to construct the detection models. Nghiem et al. [55] extracted social features (such as the total number of forum comments on the coin) from the historical social data to construct the CNN and LSTM models for detecting the Pump and dump.

5 Discussion

5.1 Challenges

Existing methods for scam detection are mainly divided into two areas: analysis-based detection and machine learning-based detection. We will discuss the challenges in the two areas respectively.

(1) Challenges in analysis-based detection

Existing analysis-based detection methods mainly use code analysis techniques, such as symbolic execution, to analyze the execution paths and behaviour of the smart contract. The detection depends on the patterns of scams, which are defined manually by collecting and analyzing the existing blockchain scams. These detection methods perform well in detecting known scams. However, when facing unknown scams, or when scammers deliberately disguise scams, the performance will be greatly affected. With the increasing number of smart contracts, it is challenging in terms of the expandability and efficiency of such methods since it requires to define new patterns for new scams in the detection.

(2) Challenges in machine learning-based detection

Machine learning-based detection methods do not require defining patterns manually and are more expandable than the analysis-based detection methods. In addition, these methods can maintain good efficiency and certain effectiveness in the face of the increasingly large amount of data on the blockchain.

For the code features based detection methods, the obvious shortcoming is that the extracted features are simple and incomplete. For example, the frequency of opcodes is commonly employed to construct the detect model in most of the works. On the one hand, the opcode frequency cannot represent the complete semantic information of the smart contract. On the other hand, scammers can manually insert redundant codes into the smart contract to obfuscate the feature difference between scams and non-scams.

For the transaction features based detection methods, the detection performance depends on the existing transaction records. And the detection for the target address also requires its transaction records. If there is no enough transaction information for the target address, the detection can not be performed. Especially for the Pump and dump, of which the research is still in the early stage, the related data is insufficient. Since the detection for Pump and dump depends on the changes in cryptocurrency prices, it is challenging to detect the Pump and dump in real time. In addition, for transaction network based detection methods, various network structures are defined and various features are extracted. However, there is no comparison with the same dataset among these methods. It is difficult to evaluate the effectiveness of these methods.

The effectiveness of machine learning-based detection methods is related to the dataset. On the blockchain platforms, a lot of scam addresses are not identified and labelled. The number of scams is small, which cause the imbalance between positive and negative samples in the dataset. In addition, with the rapid development of blockchain over the years, a significant portion of data becomes meaningless historical data, which may be still used in existing research. This will affect the efficiency in detecting future scams.

5.2 Future Directions

Based on the above discussion about the shortcomings of existing blockchain scam detection methods, this paper summarizes the following future research directions.

Constructing standard datasets for various scams. The datasets for each scam used in the existing works are different. It is necessary to construct the standard datasets for various scams, which are of large scale and contain enough scam samples, so that researchers can employ them to construct detection models and perform experimental evaluation. Existing datasets can be integrated and supplemented with the latest data on the blockchain platforms. For the data imbalance problem, on the one hand, the existing addresses on the blockchian platform can be analyzed with the existing effective scam detection methods to identify the scams; on the other hand, new scams can be created based on the defined scam patterns. With these scams integrated into the dataset, the data imbalance can be solved.

Improving the code oriented feature extraction. The existing code oriented feature extraction methods mainly focus on the frequency-related features, which are simple and incomplete. To make the machine learning based model perform better in scam detection, the training data should contain as many scam related features as possible, such as the control flow information and data flow information in the smart contract. The pattern of each scam should be analyzed to determine which elements are the scam related features. And corresponding feature extraction methods need to be studied to extract these features.

Studying a method to detect multiple scams. All the existing scam detection methods are designed to detect one type of scam. There is no detection method that can detect multiple types of scams. However, the data and features used in some detection methods for different scams may be of the same type. For example, Refs. [18] and [50] both extracted frequency features from the opcodes of smart contracts to detect Ponzi scheme and Honeypot. It is possible to design a method to detect multiple scams. With such a method, it can easily judge whether there is a scam at the target address and what type of scam it is.

6 Conclusion

This paper surveyed the existing detection methods for blockchain scams, including Ponzi scheme, Honeypot, Phishing scam, and Pump and dump. The detection methods were categorized into analysis-based methods and machine learning-based methods. More specifically, the detection methods for each scam type were summarized from various aspects, such as the type of dataset, the extracted feature, the constructed model, etc. Finally, the challenges in analysis-based detection and machine learning-based detection were analyzed respectively. And the future research directions were discussed.

Acknowledgments

The work is supported by the National Natural Science Foundation of China (No.62272145 and No.U21B2016), the Fundamental Research Funds for the Central Universities of China (B220202072, B210202075), the Natural Science Foundation of Jiangsu Province (BK20191297, BK20170893), and the CloudTech RMIT Green Bitcoin Joint Research Program/Laboratory.

References

1. Yli-Huumo, J., Ko, D., Choi, S., etc.: Where is current research on blockchain technology?—a systematic review. PloS one **11**(10) (2016)
2. CHAINALYSIS: The 2022 crypto crime report (2022), `https://go.chainalysis.com/2022-crypto-crime-report.html`
3. Bartoletti, M., Carta, S., Cimoli, T., Saia, R.: Dissecting ponzi schemes on ethereum: Identification, analysis, and impact. FGCS **102**, 259–277 (2020)
4. Torres, C.F., Steichen, M., State, R.: The art of the scam: Demystifying honeypots in ethereum smart contracts. In: USENIX Security 19. pp. 1591–1607 (2019)
5. Kumar, N., Singh, A., Handa, A., etc.: Detecting malicious accounts on the ethereum blockchain with supervised learning. In: CSCML. pp. 94–109 (2020)
6. Li, J., Gu, C., Wei, F., etc.: A survey on blockchain anomaly detection using data mining techniques. In: BlockSys. pp. 491–504 (2020)
7. Bartoletti, M., Lande, S., Loddo, A., etc.: Cryptocurrency scams: Analysis and perspectives. IEEE Access **9**, 148353–148373 (2021)

8. Kamišalić, A., Kramberger, R., Fister Jr, I.: Synergy of blockchain technology and data mining techniques for anomaly detection. APPL SCI **11**(17), 7987 (2021)
9. Wu, J., Liu, J., Zhao, Y., etc.: Analysis of cryptocurrency transactions from a network perspective: An overview. J NETW COMPUT APPL **190**, 103139 (2021)
10. Chen, W., Guo, X., Chen, Z., etc.: Phishing scam detection on ethereum: Towards financial security for blockchain ecosystem. In: IJCAI. vol. 7, pp. 4456–4462 (2020)
11. Xu, J., Livshits, B.: The anatomy of a cryptocurrency pump-and-dump scheme. In: USENIX Security Symposium. pp. 1609–1625 (2019)
12. Chen, W., Li, X., Sui, Y., etc.: Sadponzi: Detecting and characterizing ponzi schemes in ethereum smart contracts. POMACS **5**(2), 1–30 (2021)
13. Sun, W., Xu, G., Yang, Z., etc.: Early detection of smart ponzi scheme contracts based on behavior forest similarity. In: QRS. pp. 297–309 (2020)
14. Song, L., Kong, X.: A study on characteristics and identification of smart ponzi schemes. IEEE Access **10**, 57299–57308 (2022)
15. Torres, C.F., Baden, M., State, R.: Towards usable protection against honeypots. In: ICBC. pp. 1–2 (2020)
16. Ji, T., Fang, B., Cui, X., etc.: Cadetector: Cross-family anisotropic contract honeypot detection method. Chinese Journal of Computers **45**(4), 877–895 (2022)
17. Fan, S., Fu, S., Xu, H., etc.: Expose your mask: smart ponzi schemes detection on blockchain. In: IJCNN. pp. 1–7 (2020)
18. Fan, S., Fu, S., Xu, H., etc.: Al-spsd: Anti-leakage smart ponzi schemes detection in blockchain. INFORM PROCESS MANAG **58**(4), 102587 (2021)
19. Peng, J., Xiao, G.: Detection of smart ponzi schemes using opcode. In: BlockSys. pp. 192–204 (2020)
20. Lou, Y., Zhang, Y., Chen, S.: Ponzi contracts detection based on improved convolutional neural network. In: SCC. pp. 353–360 (2020)
21. Shen, X., Jiang, S., Zhang, L.: Mining bytecode features of smart contracts to detect ponzi scheme on blockchain. CMES **127**(3), 1069–1085 (2021)
22. Ibba, G., Pierro, G.A., Di Francesco, M.: Evaluating machine-learning techniques for detecting smart ponzi schemes. In: WETSEB. pp. 34–40 (2021)
23. Chen, Y., Dai, H., Yu, X., etc.: Improving ponzi scheme contract detection using multi-channel textcnn and transformer. Sensors **21**(19), 6417 (2021)
24. Bartoletti, M., Pes, B., Serusi, S.: Data mining for detecting bitcoin ponzi schemes. In: CVCBT. pp. 75–84 (2018)
25. Yu, S., Jin, J., Xie, Y., etc.: Ponzi scheme detection in ethereum transaction network. In: BlockSys. pp. 175–186 (2021)
26. Chen, W., Zheng, Z., Cui, J.: Detecting ponzi schemes on ethereum: Towards healthier blockchain technology. In: WWW 2018. pp. 1409–1418 (2018)
27. Chen, W., Zheng, Z., Ngai, E.C.H., etc.: Exploiting blockchain data to detect smart ponzi schemes on ethereum. IEEE Access **7**, 37575–37586 (2019)
28. Jung, E., Le Tilly, M., Gehani, A., etc.: Data mining-based ethereum fraud detection. In: Blockchain. pp. 266–273 (2019)
29. Wang, L., Cheng, H., Zheng, Z., etc.: Ponzi scheme detection via oversampling-based long short-term memory for smart contracts. KNOWL-BASED SYST **228**, 107312 (2021)
30. Zhang, Y., Yu, W., Li, Z., etc.: Detecting ethereum ponzi schemes based on improved lightgbm algorithm. IEEE TCSS **9**(2), 624–637 (2021)
31. He, X., Yang, T., Chen, L.: Ctrf: Ethereum-based ponzi contract identification. SECUR COMMUN NETW **2022**, 10 (2022)
32. Bian, L., Zhang, L., Zhao, K., etc.: Image-based scam detection method using an attention capsule network. IEEE Access **9**, 33654–33665 (2021)

33. Zhang, Y., Kang, S., Dai, W., etc.: Code will speak: Early detection of ponzi smart contracts on ethereum. In: SCC. pp. 301–308 (2021)
34. Aljofey, A., Jiang, Q., Qu, Q.: A supervised learning model for detecting ponzi contracts in ethereum blockchain. In: ICBDS. pp. 657–672 (2021)
35. Jin, C., Jin, J., Zhou, J., etc.: Heterogeneous feature augmentation for ponzi detection in ethereum. IEEE T CIRCUITS-II **69**(9), 3919–3923 (2022)
36. Fan, S., Xu, H., Fu, S., etc.: Smart ponzi scheme detection using federated learning. In: HPCC/SmartCity/DSS. pp. 881–888 (2020)
37. Yuan, Q., Huang, B., Zhang, J., etc.: Detecting phishing scams on ethereum based on transaction records. In: ISCAS. pp. 1–5 (2020)
38. Yuan, Z., Yuan, Q., Wu, J.: Phishing detection on ethereum via learning representation of transaction subgraphs. In: BlockSys. pp. 178–191 (2020)
39. Wu, J., Yuan, Q., Lin, D., etc.: Who are the phishers? phishing scam detection on ethereum via network embedding. IEEE T SYST MAN CY-S **52**(2), 1156–1166 (2020)
40. Tang, J., Zhao, G., Zou, B.: Semi-supervised graph convolutional network for ethereum phishing scam recognition. In: ECNCT. vol. 12167, pp. 369–375 (2022)
41. Yu, T., Chen, X., Xu, Z., etc.: Mp-gcn: A phishing nodes detection approach via graph convolution network for ethereum. APPL SCI **12**(14), 7294 (2022)
42. Wan, Y., Xiao, F., Zhang, D.: Early-stage phishing detection on the ethereum transaction network. Soft Computing **27**(7), 3707–3719 (2023)
43. Wen, H., Fang, J., Wu, J., etc.: Transaction-based hidden strategies against general phishing detection framework on ethereum. In: ISCAS. pp. 1–5 (2021)
44. Chen, L., Peng, J., Liu, Y., etc.: Phishing scams detection in ethereum transaction network. TOIT **21**(1), 1–16 (2020)
45. Wang, J., Chen, P., Yu, S., etc.: Tsgn: Transaction subgraph networks for identifying ethereum phishing accounts. In: BlockSys. pp. 187–200 (2021)
46. Xia, Y., Liu, J., Wu, J.: Phishing detection on ethereum via attributed ego-graph embedding. IEEE T CIRCUITS-II **69**(5), 2538–2542 (2022)
47. Li, S., Gou, G., Liu, C., etc.: Ttagn: Temporal transaction aggregation graph network for ethereum phishing scams detection. In: WWW. pp. 661–669 (2022)
48. Fu, B., Yu, X., Feng, T.: Ct-gcn: a phishing identification model for blockchain cryptocurrency transactions. INT J INF SECUR **21**(6), 1–10 (2022)
49. Zhang, D., Chen, J., Lu, X.: Blockchain Phishing scam detection via multi-channel graph classification. In: BlockSys. pp. 241–256 (2021)
50. Chen, W., Guo, X., Chen, Z.: Honeypot contract risk warning on ethereum smart contracts. In: JCC. pp. 1–8 (2020)
51. Hara, K., Takahashi, T., Ishimaki, M., etc.: Machine-learning approach using solidity bytecode for smart-contract honeypot detection in the ethereum. In: QRS-C. pp. 652–659 (2021)
52. Camino, R., Torres, C.F., Baden, M., etc.: A data science approach for detecting honeypots in ethereum. In: ICBC. pp. 1–9 (2020)
53. La Morgia, M., Mei, A., Sassi, F., etc.: Pump and dumps in the bitcoin era: Real time detection of cryptocurrency market manipulations. In: ICCCN. pp. 1–9 (2020)
54. Morgia, M.L., Mei, A., Sassi, F., etc.: The doge of wall street: Analysis and detection of pump and dump cryptocurrency manipulations. ACM T INTERNET TECHN **23**(1), 1–28 (2023)
55. Nghiem, H., Muric, G., Morstatter, F., etc.: Detecting cryptocurrency pump-and-dump frauds using market and social signals. EXPERT SYST APPL **182**, 115284 (2021)