INTERNATIONAL JOURNAL OF SOFTWARE ENGINEERING & COMPUTER SYSTEMS (IJSECS)
ISSN: 2289-8522 e-ISSN: 2180-0650
VOL. 8, ISSUE 2, 43 – 50
DOI: https://doi.org/10.15282/ijsecs.8.2.2022.5.0102

ORIGINAL ARTICLE

Fraudulent Account Detection in the Ethereum’s Network Using Various Machine Learning Techniques

Amer Sallam[1*], Taha H. Rassem[2], Hanadi Abdu[1], Haneen Abdulkareem[1], Nada Saif[1] and Samia Abdullah[1]

[1]Faculty of Engineering and IT, Taiz University, Yemen
[2]Faculty of Science and Technology, Bournemouth University, United Kingdom

ABSTRACT – On the Ethereum network, users communicate with one another through a variety of different accounts. Pseudo-anonymity was enforced over the network to provide the highest level of privacy. By using accounts that engage in fraudulent activity across the network, such privacy may be exploited. Like other cryptocurrencies, Ethereum blockchain may exploited with several fraudulent activities such as Ponzi schemes, phishing, or Initial Coin Offering (ICO) exits, etc. However, the identification of parameters with abnormal account characteristics is not an easy task and requires an intelligent approach to distinguish between normal and fraudulent activities. Therefore, this paper has attempted to solve this ~~a~~ problem by using machine learning techniques to introduce a robust approach that can detect fraudulent accounts on Ethereum. We have used a K-Nearest Neighbor, Random Forest and XGBoost over a collected dataset of 4,681 instances along with 2,179 fraudulent accounts associated and 2,502 regular accounts. The XGBoost, RF, and KNN techniques achieved average accuracies of 96.80 %, 94.8 8%, and 87.85% and an average AUC of 0.995, 0.99 and 0.93, respectively.

ARTICLE HISTORY
Received: 1 May 2022
Revised: 7 August 2022
Accepted: 24 August 2022

KEYWORDS
Ethereum
Fraudulent Accounts
Machine Learning

INTRODUCTION

The new era of technology has brought various changes in the ways of using data. Smartphones, social networks, cloud platforms, and the Internet of Things (IoT) are leading to new innovations and close relations between customers. To revolutionize this integration of data and customer demands, many architectures have been designed to enrich the user experience and provide better-informed decisions. The continuous growth of data in networks brings security challenges in a distributed environment. Thus, new dimensions to the systems’ security and efficiency are needed. The combination of cryptography and distributed ledgers have made a new class of technology called Blockchain [1]. Blockchain network is composed of a set of peers that collaborate to ensure the security of a distributed database (ledger). Although Blockchain technology was initially introduced in 2008 by Satoshi Nakamoto as a Bitcoin cryptocurrency [2], it currently uses a distributed network for exchanging any service or transaction securely [3]. Blockchain technology was widely integrated into several domains such as healthcare [4], E-Voting [5], migrants' remittances [6], online payment [7], IoT [8], smart contracts for insurance [9], digital assets and author royalty protecting [10], educational documents storing and authentication [11], tracking tangible asset ownership [12], and intellectual property rights [13], etc.

Blockchain is a promising technology [14] but still faced many challenges, such as malicious user who choose to act fraudulently to gain greater rewards than that gained when they act fairly [15]. Ethereum has been introduced to add another layer of programmability to the blockchain. Ethereum’s users interact with each other in the Ethereum network via different accounts. Pseudo-anonymity was enforced over the network to provide high privacy. This privacy has been exploited by accounts carrying fraudulent behavior over the network. Attempting to identify parameters that exhibit abnormal characteristics of these users manually is difficult due to the architectural nature of the distributed ledger. Thus, the implementation of machine learning (ML) algorithms on such a network is required to distinguish between transactions that act normally or with those with fraudulent behavior among user accounts, through learning features related to either normal or fraudulent behavior [16].

This paper tries to tackle such problems by using various ML techniques for detecting and identifying fraud accounts over the Ethereum Blockchain network. The rest of the paper is organized as follows: The background section provides an overview of blockchain technology, the related work section discusses the related works and the role of machine learning algorithms in the detection of fraud accounts, methodology section introduces the methodology and the selected techniques used in the study, section results and discussion presents the results and the evaluation process, and conclusion section summarizes the work presented in the paper and mentions the future works.

*CORRESPONDING AUTHOR | Amer Sallam | ✉ amer.sallam@gmail.com

BACKGROUND

Blockchain has been considered as a sub-category of Distributed Ledger Technology (DLT) [17]. DLT is an umbrella term used to describe technologies of recording and sharing a distributed ledger collectively maintained and controlled by a distributed network of nodes, either privately or publicly, of which each node has exact copy of data records [15]. The basic element in blockchain architecture, as illustrated in Figure 1, is the transaction. Transactions go through a validation process and broadcast, forming a block of transactions. A consensus process is applied on blocks to select the next block to be added to a chain of blocks. In the generated list of blocks, each block is linked to the one before it. Consensus of blocks and validation of transaction processes are carried out by special nodes in blockchain, called miners [18].

Ethereum blockchain [19] was introduced in 2013 and its platform was launched in 2015 [20]. Ethereum adds another layer of programmability on blockchain, that is the most widely used Decentralized Applications (DApp) development platform based on the blockchain, through a Turing-complete programming language support called solidity [21], which can be used to write a smart contract. Smart contracts are embedded scripts in blockchain with a unique address, enabling operators and providers to specify their conditions, business rules, and sanctions. The term “Ethereum” can be used to refer to three distinct things: the Ethereum protocol, the Ethereum network created by computers using the protocol, and the Ethereum project funding development of the aforementioned two [22]. Ethereum protocol is widely known as a development of Bitcoin protocol, popularizing its core ideas and enabling building various applications on top of the blockchain technology [23]. Ethereum has two main components, First one is *Turing-complete virtual processor*, which called Ethereum Virtual Machine (EVM) and can execute scripts, *Token* is the second component, which is the currency used in the network for Ethereum it is called ether.

Ethereum’s users interact with each other through two types of accounts: *Externally Owned Accounts (EOA)* which are controlled by private keys, and *Contract Accounts (CA)* which are controlled through their contract code. Each account holds four attributes: Storage, Contract Code Hash, Ether Balance, and Nonce. Nonce attribute describes the quality of transactions issued by an account or a contract. Ether Balance attribute defines the available owned balance of account in Wei unit. The Contract Code Hash attribute pertains to the EVM hash code. Storage attribute is corresponding to the 256 bit hash of the Merkel Patricia tree’s root.

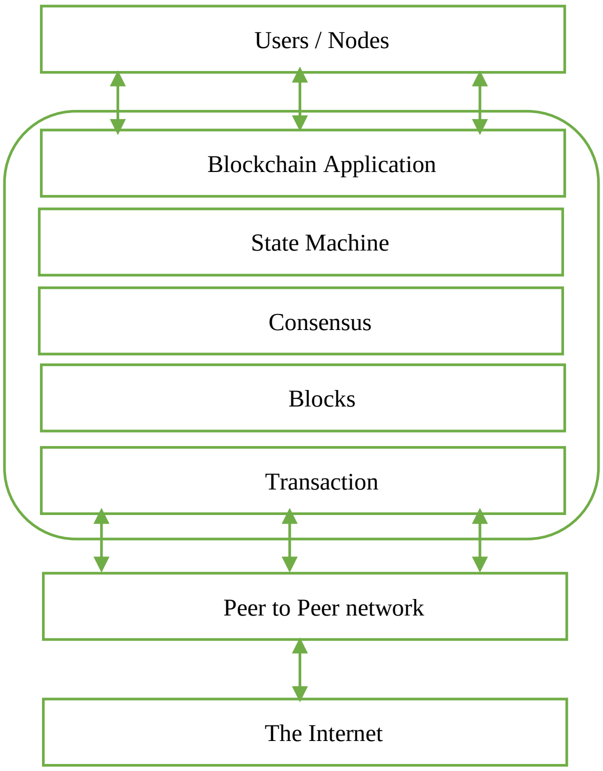

Figure 1. Foundation of Blockchain Architecture

RELATED WORK

The integration of blockchain and ML techniques have been proved as a powerful solution for many applications as stated by many researchers. Some of this research is discussed in the following section.

Carlin D. et al. [24] used Random Forest algorithm to build ML-based model that is able to distinguish between cryptomining-enabled and cryptomining-deactivated on HTML files .The main achievement of this work was the ability to classify the benign from the malicious files with high accuracy 99.9%.

Chen W. et al. [25] used eXtreme Gradient Boosting algorithm to detect smart Ponzi schemes based on the extracted features. They conducted their experiments on three types of features; account, opcode, and the combination of both.

The Experimental results showed that the model's performance improved and achieved an F1-score of 87% while combining opcode and account features.

Jung E. et al. [26] tried to build a data mining-based model that serves as the first defense line against Ponzi schemes using several classification algorithms such as J48, and Random Forest with a precision of 99% and recall of 97%.

Baek H. et al. [27] proposed a model to investigate the cryptocurrency wallets and detect the fraudulent parties using K-means, Random Forest (RF), and Expectation Maximization (EM) algorithms. They obtained F1-score of 96% to detect cryptocurrency wallets with fraudulent behavior.

Poursafaei F. et al. [28] proposed a model to detect malicious entities in Ethereum network, applying Random Forest, Logistic Regression, AdaBoost, and Support Vector Machine classification methods with F1-score of 99%.

Weili C. et al. [29] proposed a systematic approach to detect phishing blockchain accounts. Support vector machine (SVM), Decision Tree (DT), light gradient boosting (lightGBM) and, DElightGBM algorithms were used. Extensive experiments showed that the DElightGBM algorithm could effectively identify phishing scams with an F1-score of 81.22%.

Lei W. et al. [30] proposed an approach based on oversampling-based Long Short-Time Memory (LSTM) to detect Ponzi schemes in Ethereum called PSD-OL. PSD-OL approach combines oversampling with LSTM and takes both contract account features and contract code features in consideration. Experimental results showed that their approach has achieved F1-score of 96%.

Huiwen H. et al. [31] presented a deep learning-based scam detection framework for Ethereum networks. They designed a GUR network that can determine whether a smart contract is fraudulent or not by learning from the N-gram bytecode patterns, with 97% F1-score.

The above-mentioned related work demonstrates the effectiveness of solutions based on machine learning and deep learning techniques in improving the Ethereum security. However, most of these works have focused only on one type of fraud, while in this paper, various ML techniques have been used to present a proposed solution that can discover different types of fraudulent accounts, by analysing the patterns of regular and fraudulent accounts.

METHODOLOGY

In this framework, different steps, as depicted in Figure 2, are performed to detect the fraud accounts in Ethereum's network.

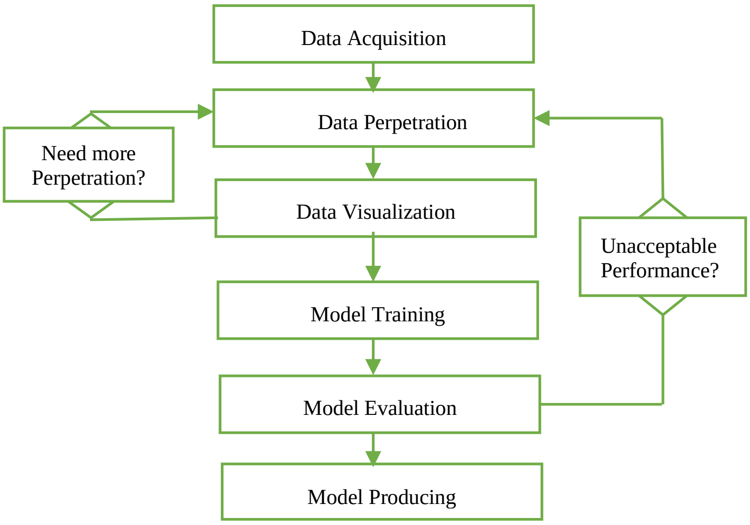

Figure 2. Methodology Flowchart

Dataset Acquisition

Dataset can be scraped from various web servers' logs or census datasets using third party APIs. However, the Ethereum fraud detection dataset [16] which consists of 4,681 instances along with 2,179 fraudulent accounts associated and 2,502 normal accounts with a total of 42 features per instance was used in this work. The extracted features include average time between sent/received transaction in minutes, time difference between first and last transaction, total number of sent/received normal transactions, min/max value of Ether ever sent/received, average value in Ether ever sent/received, and so on.

Data Cleansing

Data cleansing is the process of combining, structuring, and organizing data. Data usually have a lot of inconsistencies such as missing values, empty columns, and disproportionate data format. For this reason, data needs to be processed, explored, and conditioned before building the required model. Data cleansing also involves finding the relevant data that needs to be included in analytics to ensure delivering the information that the analysts are seeking for. Omitted values, duplicate examples, bad labels, and bad feature values are some reasons for data unreliability. Therefore, the dataset was prepared by handling Not a Number (NAN) values and delete the less informative features such as: Index, Address, ERC20_most_sent_token_type, ERC20_most_rec_token_type, ERC20_uniq_sent_token_name, ERC20_uniq_ rec_token_nam, and so on.

Data Visualization

Visualization plays a vital role in commutating linguistic theory, due to the highly visible nature of the human mind. There are various ways for visualizing the data. Given that the task in hand contains 42 dimensions, a powerful solution is to use dimension reduction methods. In dimensionality reduction methods, the high-dimensional of dataset X={x_1, x_2, … ,x_n} is converted into two-dimensional (2D) or three-dimensional (3D) data, which can be figured in a scatterplot [32]. One of the most famous form of dimensionality reduction of is t-Distributed Stochastic Neighbour Embedding (t-SNE). t-SNE is a nonlinear algorithm, that allows to blow up the densest vector conglomerates and shorten the distances between the most remote regions.

Model Building

This is the stage where the machine learning algorithms are trained. The built models undergo the following procedure in order to produce the desired output:

1) Split dataset into train set (90%) and test set (10%)
2) Train the defined classification model on the training set
3) Utilize the trained model to predict on the corresponding test set and output results

The following ML algorithms were used in this work:

a) Extreme Gradient Boosting Algorithm (XGBoost) as defined in [33], is a scalable ML system for tree boosting. XGBoost belongs to an ensemble learning class based on gradient boosting algorithms [34]. XGBoost focuses on reducing the best split's computational complexity, which is the most time-consuming process in decision trees. So, it can solve problems using a minimal number of resources. Mathematically, the XGBoost model can be in the following form:

$$\widehat{Y_\iota} = \sum_{k=1}^{K} f_k(X_i), f_k \in F \tag{1}$$

Where, K is the number of trees, f is the functional space of F, F is the set of possible classification and regression trees (CARTs).

b) Random Forest (RF) as defined in [35], is also an ensemble algorithm, RF classifier creates a set of decision trees (DT) from randomly selected subsets of the training set. Votes generated from different decision trees are aggregated to decide the final class of test data.

c) K-Nearest Neighbor (KNN) is a simple and effective supervised algorithm used for regression/classification [36]. In the case of classification, KNN calculates the distance between the test data and all the training points, forming a neighborhood of test data. The most popular distance measure is Euclidean distance, which equals the square root of the sum of squared difference between a new point (X) and existing point (Xi) across all input attributes j.

$$\text{Euclidean distance} = \sqrt[2]{sum((Xj - Xij)^2)} \tag{2}$$

Then the K nearest points to the test data are selected. KNN algorithm calculates probability of test data belonging to the classes of K training data and selects the class with high probability. Class probability can be calculated as following:

$$p(class = 0) = \frac{count(class=0)}{count(class=0)+count(class=1)} \tag{3}$$

KNN's performance depends on k value. Therefore, choosing the best k value is an important issue.

Performance Evaluation

Conceptually, binary classification can be considered as the most common and simple application of ML. However, there are still several issues for evaluating this simple task [37]. Evaluating a classifier is often easier than evaluating a regressor [38]. Followings are some of performance measures used in binary classification:

1) Confusion matrix is a comprehensive way to evaluate a classifier. In the case of binary classification, Confusion matrix's output is a square array, where rows and columns refer to the actual and predicted classes respectively.
   - True Positive (TP): The fraudulent accounts classified as fraudulent accounts.
   - False Positive (FP): The unfraudulent accounts classified as fraudulent accounts.

- False Negative (FN): The fraudulent accounts classified as unfraudulent accounts.
- True Negative (TN): The unfraudulent accounts classified as unfraudulent accounts.

Confusion matrix can be summarized in several ways, which can be expressed as following:

Accuracy, is the number of correct predictions per total available samples being tested.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{4}$$

Precision is a measure of the actual positive samples predicted as positive.

$$\text{Precision} = \frac{TP}{TP+FP} \tag{5}$$

Sensitivity, or as also commonly known as Recall and True Positive Rate (TPR), allows us to determine the rate of real positives that are correctly identified.

$$Sensitivity = \frac{TP}{TP+FN} \tag{6}$$

In this study, we are mostly interested in ensuring that fraudulent accounts are detected without classifying a normal account as a fraudulent account; therefore, F1-score measure was selected (XGBoost model has the best recall value). F-measure is the weighted average of recall and precision. It is also commonly known as f-score.

$$F - measure = 2 \times \frac{recall \times precision}{recall+precision} \tag{7}$$

2) Receiver Operating Characteristic (ROC) curve is a common measure for a classifier's performance. It is commonly used to analyze the behavior of classifiers at different thresholds. It considers all possible thresholds for a classifier, showing false positive rate (FPR) against true positive rate (TPR). False positive rate is the fraction of false positives out of all negative samples:

$$FPR = \frac{FP}{FP+TN} \tag{8}$$

ROC curve needs to be summarized using a single number, this value commonly refers to Area Under Curve (AUC).

RESULTS AND DISCUSSION

The Figure 3 and Figure 4 show the 2D and 3D scatter plots respectively using t-SNE to help visualize the available dataset with respect to the account class. The red data points indicate fraudulent accounts while the blue data points refer to the normal accounts. Although there are a few distinguishable clusters from both classes, there is a noticeable overlap between the data points. Both figures confirm the high level of impurity and it can be understood that two classes are not linearly separable. Therefore, it is necessary to use ML classifiers to find the designated features among these classes.

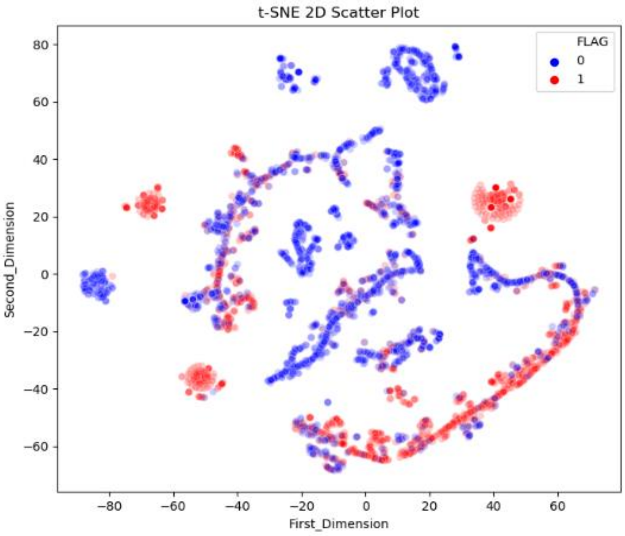

Figure 3. 2D scatter plot of the dataset

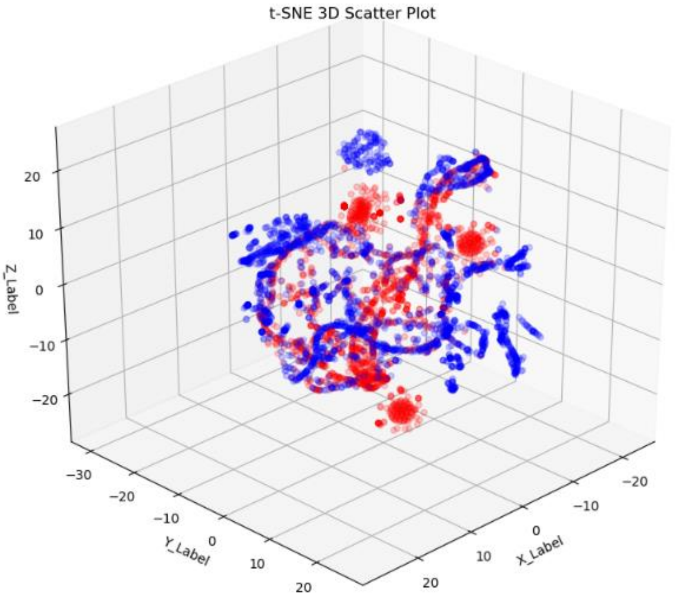

Figure 4. T-SNE three-dimentional scatter plot

Three different classifiers i.e., KNN, RF, XGBoost have been used in this work to identify such features in order to detect and classify the fraud account and to obtain more accurate results of detection due to their efficiencies and auto learning abilities. Extensive experiments were carried out by each classifier using fraud detection dataset [16] that contains more than 40 features and define various types of fraudulent accounts.

The performance of each classifier is evaluated based on Accuracy, Precision, Recall and F1-score. The results of evaluation metrics are presented in Table 1, the results have shown that the XGBoost classifier is the most successful detector for the fraud account with 96.80 % of accuracy and 96% F1-score, the RF achieved also a plausible result with 94.8 8%, accuracy and 95% for F1-score, while the KNN obtained 87.85% and 88% for accuracy and F1-score respectively.

Table 1. Models Evaluation.

Model Name	Accuracy	Precision	Recall	F1-score	Execution Time (Sec.)
XGBoost	**96.80**	**96.50**	**96.00**	**96.50**	1.232
Random Forest	94.88	95.00	94.5	95.00	3.268
K-Nearest Neighbor	87.85	88.00	87.50	88.00	**0.206**

The above results were achieved after numerous experiments along with tuning some hyperparameters as depicted in Table 2:

Table 2. The tunned hyperparameters

Classifier	Hyperparameter
XGBoost	Maximum Tree Depth = 3 Number Of Trees = 200

Random Forest	N_Estimators = 600
	Min_Samples_Split = 5
	Min_Samples_Leaf = 2

K-nearest Neighbor	K = 6

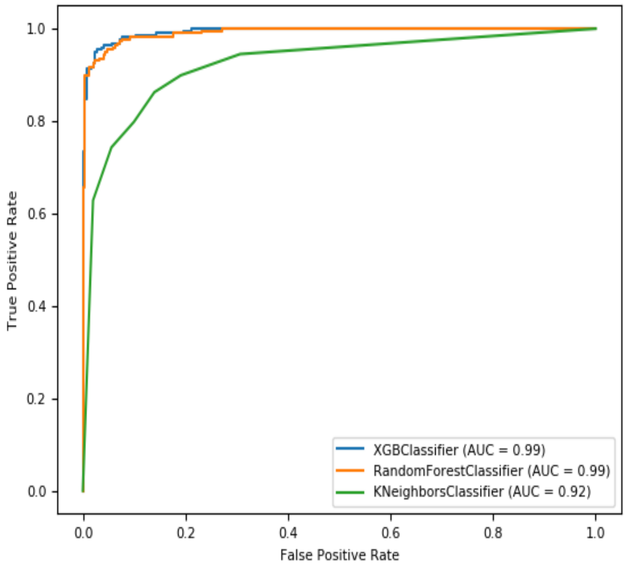

Figure 5. AUC performance of the different models.

The results of AUC shown in Figure 5 have also indicated that the XGBoost classifier can perform well compared to other classifiers for detecting the fraudulent account over the Ethereum's Network.

CONCLUSION

The continuous growth of data ~~among~~ between the networks brings security challenges in a distributed environment. Thus, new dimensions of security and efficiency are needed. The combination of cryptography and distributed ledgers have made a new class of technology called Blockchain [1]. Blockchain network is composed of a set of peers that collaborate to ensure the security of a distributed database (ledger). It currently uses a distributed network for exchanging any service or transaction securely. Ethereum has been introduced to add another layer of programmability to the blockchain. Ethereum networks still suffered from fraudulent activities which reduce the trust between users. In this paper, a solution based on various ML techniques for detecting and identifying the patterns of the different types of fraudulent accounts in Ethereum's network was proposed. A simple illustration of the detection phases was explained.

Tests on instances obtained from a fraud detection dataset that contains more than 40 features per instance ~~which~~ that define various types of fraudulent accounts are conducted and experiments using 4,681instances are carried out. Three different classifiers i.e., KNN, RF, and XGBoost are used to classify such features to detect and identify the fraud account and to obtain more accurate results of detection due to their efficiencies and auto-learning abilities. The performance of each classifier is evaluated based on Accuracy, Precision, Recall and F1 score. The results show that XGBoost is the most successful detector for the fraud account with 96.80 % of accuracy, while RF and KNN also achieved plausible accuracy results with 94.88%, and 87.85% respectively. However, the obtained results were comprehensive, demonstrating the significance of the feature extraction process and its implications for the rest of the fraud detection system. However, the dataset used in this study was limited in size and scope and needs to be improved with a greater volume of accounts and corresponding features. We look forward to generalizing the proposed solution to all other blockchain networks in the future as this work was limited to Ethereum.

REFERENCES

[1] Sapra, Riya, and Parneeta Dhaliwal. "Blockchain: the new era of technology." In 2018 fifth international conference on parallel, distributed and grid computing (PDGC), pp. 495-499. IEEE, 2018.

[2] Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." Decentralized Business Review (2008): 21260.

[3] Ahram, Tareq, Arman Sargolzaei, Saman Sargolzaei, Jeff Daniels, and Ben Amaba. "Blockchain technology innovations." In 2017 IEEE technology & engineering management conference (TEMSCON), pp. 137-141. IEEE, 2017.

[4] Ekblaw, Ariel, Asaph Azaria, John D. Halamka, and Andrew Lippman. "A Case Study for Blockchain in Healthcare:"MedRec" prototype for electronic health records and medical research data." In Proceedings of IEEE open & big data conference, vol. 13, p. 13. 2016.

[5] Yavuz, Emre, Ali Kaan Koç, Umut Can Çabuk, and Gökhan Dalkılıç. "Towards secure e-voting using ethereum blockchain." In 2018 6th International Symposium on Digital Forensic and Security (ISDFS), pp. 1-7. IEEE, 2018.

[6] Massimo FLORE, How Blockchain-Based Technology Is Disrupting Migrants' Remittances: A Preliminary Assessment, EUR 29492 EN, Publications Office of the European Union, Luxembourg, 2018.

[7] Jamil, Faisal, Omar Cheikhrouhou, Harun Jamil, Anis Koubaa, Abdelouahid Derhab, and Mohamed Amine Ferrag. "PetroBlock: A blockchain-based payment mechanism for fueling smart vehicles." Applied Sciences 11, no. 7 (2021): 3055.

[8] Pranav Ratta, Amanpreet Kaur, Sparsh Sharma, Mohammad Shabaz, Gaurav Dhiman, "Application of Blockchain and Internet of Things in Healthcare and Medical Sector: Applications, Challenges, and Future Perspectives", *Journal of Food Quality*, vol. 2021, Article ID 7608296, 20 pages, 2021

[9] Kar, Arpan Kumar, and L. Navin. "Diffusion of blockchain in insurance industry: An analysis through the review of academic and trade literature." Telematics and Informatics 58 (2021): 101532.

[10] Guo, J., Li, C., Zhang, G. *et al.* Blockchain-enabled digital rights management for multimedia resources of online education. *Multimed Tools Appl* **79**, 9735–9755 (2020)

[11] Guustaaf, Edward, Untung Rahardja, Qurotul Aini, Herliana Wahyu Maharani, and Nesti Anggraini Santoso. "Blockchain-based Education Project." *Aptisi Transactions on Management (ATM)* 5, no. 1 (2021): 46-61.

[12] Adams, R, Parry, G, Godsiff, P, Ward, P. The future of money and further applications of the blockchain. *Strategic Change*. 2017; 26: 417– 422.

[13] Christian Catalini and Joshua S. Gans. 2020. "Some simple economics of the blockchain," Communication of the ACM 63, 7 (July 2020), 80–90.

[14] Li, Xiaoyun, Zibin Zheng, and Hong-Ning Dai. "When services computing meets blockchain: Challenges and opportunities." Journal of Parallel and Distributed Computing 150 (2021): 1-14.

[15] Lim, Ming K., Yan Li, Chao Wang, and Ming-Lang Tseng. "A literature review of blockchain technology applications in supply chains: A comprehensive analysis of themes, methodologies and industries." Computers & Industrial Engineering 154 (2021): 107133.

[16] Farrugia, Steven, Joshua Ellul, and George Azzopardi. "Detection of illicit accounts over the Ethereum blockchain." *Expert Systems with Applications* 150 (2020): 113318.

[17] European Parliament, Directorate-General for Internal Policies of the Union, Snyers, A., Houben, R. (2018). *Cryptocurrencies and blockchain : legal context and implications for financial crime, money laundering and tax evasion*, European Parliament.

[18] Valenta, Martin, and Philipp Sandner. "Comparison of ethereum, hyperledger fabric and corda." *Frankfurt School Blockchain Center* 8 (2017): 1-8.

[19] Buterin, Vitalik. "A next-generation smart contract and decentralized application platform." *white paper* 3, no. 37 (2014).

[20] Wood, Gavin. "Ethereum: A secure decentralised generalised transaction ledger." *Ethereum project yellow paper* 151, no. 2014 (2014): 1-32.

[21] Vujičić, Dejan, Dijana Jagodić, and Siniša Ranđić. "Blockchain technology, bitcoin, and Ethereum: A brief overview." In *2018 17th international symposium infoteh-jahorina (infoteh)*, pp. 1-6. IEEE, 2018

[22] Dannen, Chris. *Introducing Ethereum and solidity*. Vol. 1. Berkeley: Apress, 2017.

[23] Wang, Shuai, Liwei Ouyang, Yong Yuan, Xiaochun Ni, Xuan Han, and Fei-Yue Wang. "Blockchain-enabled smart contracts: architecture, applications, and future trends." *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 49, no. 11 (2019): 2266-2277

[24] Carlin, Domhnall, Philip O'kane, Sakir Sezer, and Jonah Burgess. "Detecting cryptomining using dynamic analysis." In *2018 16th Annual Conference on Privacy, Security and Trust (PST)*, pp. 1-6. IEEE, 2018.

[25] Chen, Weili, Zibin Zheng, Jiahui Cui, Edith Ngai, Peilin Zheng, and Yuren Zhou. "Detecting ponzi schemes on ethereum: Towards healthier blockchain technology." In *Proceedings of the 2018 world wide web conference*, pp. 1409-1418. 2018.

[26] Jung, Eunjin, Marion Le Tilly, Ashish Gehani, and Yunjie Ge. "Data mining-based ethereum fraud detection." In *2019 IEEE International Conference on Blockchain (Blockchain)*, pp. 266-273. IEEE, 2019

[27] Baek, Hyochang, Junhyoung Oh, Chang Yeon Kim, and Kyungho Lee. "A model for detecting cryptocurrency transactions with discernible purpose." In *2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 713-717. IEEE, 2019

[28] Poursafaei, Farimah, Ghaith Bany Hamad, and Zeljko Zilic. "Detecting Malicious Ethereum Entities via Application of Machine Learning Classification." In *2020 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS)*, pp. 120-127. IEEE, 2020

[29] Chen, Weili, Xiongfeng Guo, Zhiguang Chen, Zibin Zheng, and Yutong Lu. "Phishing Scam Detection on Ethereum: Towards Financial Security for Blockchain Ecosystem." In *IJCAI*, pp. 4506-4512. 2020

[30] Wang, Lei, Hao Cheng, Zibin Zheng, Aijun Yang, and Xiaohu Zhu. "Ponzi scheme detection via oversampling-based Long Short-Term Memory for smart contracts." *Knowledge-Based Systems* 228 (2021): 107312

[31] Hu, Huiwen, Qianlan Bai, and Yuedong Xu. "Scsguard: Deep scam detection for ethereum smart contracts." In *IEEE INFOCOM 2022-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 1-6. IEEE, 2022.

[32] Van der Maaten, Laurens, and Geoffrey Hinton. "Visualizing data using t-SNE." *Journal of machine learning research* 9, no. 11 (2008).

[33] Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785-794. 2016

[34] Bentéjac, Candice, Anna Csörgő, and Gonzalo Martínez-Muñoz. "A comparative analysis of gradient boosting algorithms." *Artificial Intelligence Review* 54, no. 3 (2021): 1937-1967

[35] Breiman, Leo. "Random forests." Machine learning 45, no. 1 (2001): 5-32.

[36] Hand, David J. "Principles of data mining." Drug safety 30, no. 7 (2007): 621-622.

[37] Müller, Andreas C., and Sarah Guido. Introduction to machine learning with Python: a guide for data scientists. " O'Reilly Media, Inc.", 2016.

[38] Géron, Aurélien. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. " O'Reilly Media, Inc.", 2019.