https://doi.org/10.7840/kics.2023.48.7.875

Malicious Account Classification Using CNN for Ethereum Blockchain's Accounts

Revin Naufal Alief◆, Syifa Maliah Rachmawati*, Jae Min Lee*, Dong-Seong Kim°

ABSTRACT

The use of cryptocurrencies for transactions has grown over the past few years. Today, cryptocurrency is the most widely used and rapidly expanding currency in the global financial market. This increase can be attributed to blockchain networks, which offer transparent and secure transactions and record cryptocurrency transactions. However, as the volume of transactions increases, fraud also surfaces, resulting in significant losses for the Ethereum account holders involved. Machine learning has been used to address this issue in a previous study; however, the study only provided a limited set of performance metrics. In this study, a CNN-based algorithm is proposed to identify fraudulent accounts in the Ethereum network. The CNN model is applied to a dataset that includes legitimate and fraudulent transactions over the Ethereum network. The results reveal that the CNN-based model successfully identified fraudulent accounts with an accuracy of 98.67%.

Key Words : Blockchain, Deep Learning, Ethereum Network, Ethereum Account, Fraud Detection.

Ⅰ. Introduction

Cryptocurrencies have attracted considerable attention in recent years. This attention is demonstrated by the increasing number of cryptocurrencies and the increasing volume of transactions in the cryptocurrency market. Customer perception of cryptocurrencies is no longer merely based on investment excitement, but also serves as evidence of stable and long-term investment[1,2]. The evidence of stable and long-term investment is shown by the absence of third parties, resulting in the quantity supply not being manipulated, in contrast to fiat currencies that are vulnerable to inflation[3]. The absence of third parties such as financial institutions also enables cryptocurrency transactions to be secure and ensures privacy, as no party controls its own funds and personal identities are not exposed[4]. Furthermore, financial institutions still use centralized systems to accommodate the needs of their customers[5], which are prone to various security risks[6].

Ethereum, one of the most popular cryptocurrencies, currently has a large transaction volume on its networks. Unfortunately, some problems still exist, particularly phishing. Since 2017, phishing accounts for 50% of all cybercrimes on Ethereum. Thus, a system for detecting fraudulent accounts is required to prevent this problem.

Artificial intelligence (AI), an emerging technology, can classify data by learning from previous data. AI has also been combined with the blockchain application field. To improve systems such

※ This research work was supported by Priority Research Centers Program through NRF funded by MEST (2018R1A6A1A03024003) and the Grand Information Technology Research Center support program (IITP-2023-2020-0-01612) supervised by the IITP by MSIT, Korea and Basic Reserch project (NRF-2022R1I1A3071844).
◆ First Author : Kumoh National Institute of Technology, Department of IT Convergence Engineering, revinnaufal@kumoh.ac.kr, 학생회원
° Corresponding Author : Kumoh National Institute of Technology, Department of IT Convergence Engineering, dskim@kumoh.ac.kr, 종신회원
* Kumoh National Institute of Technology, Department of IT Convergence Engineering, syifamr@kumoh.ac.kr, 학생회원 ljmpaul@kumoh.ac.kr, 종신회원
논문번호 : 202303-052-0-SE, Received March 20, 2023; Revised Arpil 10, 2023; Accepted Arpil 18, 2023

as healthcare[8,9], unmanned aerial vehicles[10,11], audio copyright protection[12,13], and blockchain consensus performance[14]. AI, particularly Deep Learning (DL), can classify fraud based on the transaction history. Compared to Machine Learning (ML), DL is considered a more advanced AI technique because it is less dependent on human interference.

A high accuracy in fraud-account classification can be achieved by creating a well-designed DL model. In addition, preprocessing should be performed carefully to enable the DL model to learn more accurately. Previous studies have proposed ML-based fraudulent account classifications[15,16]. However, if the ML algorithm returns inaccurate predictions, humans must still intervene to solve the problem. By contrast, DL models enable algorithms to determine the accuracy of predictions using neural networks. The main contributions of this study are as follows:

1. A DL model that uses a convolutional neural network (CNN) algorithm to classify a fraudulent account based on its transaction history is proposed.
2. The Synthetic Minority Over-sampling TEchnique (SMOTE) algorithm is applied for preprocessing to handle imbalanced data to enable the DL model to learn more accurately.

The remainder of this paper is organized as follows: Section II discusses previous studies on Ethereum's fraudulent account classification. In Section III, the proposed system model is described in detail. Sections IV and V provide a performance evaluation of the compared models as well as the conclusion of this study.

Ⅱ. Related Works

Several researches have been done to solve the problem of fraud accounts in Ethereum blockchain. The approaches have been made by implementing AI to classify the fraud account based on existing datasets. Various approaches are also considered to improve the performance of the AI model. The approach is by also considers the data preprocessing to have higher accuracy. This section discusses some previous studies, especially regarding the AI model implementation in fraud detection.

The authors in [17] developed an ML model to classify phishing attacks. Decision Tree (DT) and Random Forest (RF) algorithms are applied in this paper. Through public datasets, this paper's models show that the RF algorithm is more suitable than the DT algorithm. In addition, this paper also considered applying a feature selection algorithm so the models have an improvement in time measurement.

A. Maurya[15] proposed an ML approach for classifying fraudulent transactions using the Ethereum dataset. This paper's main focus is comparing various ML algorithms to find the most suited algorithm for classifying fraud accounts. The ML algorithms that are compared in this paper are Logistic Regression, Decision Tree, Random Forest, and XGBoost. This paper also considers the imbalanced dataset problem so that the model could learn more accurately. The result shows that XGBoost algorithm is suited for classifying fraud accounts.

R. F. Ibrahim[16] investigated illicit accounts on Ethereum blockchain and proposed a fraud detection model using three ML algorithms. These three algorithms are Decision Tree, Random Forest, and K-Nearest Neighbour. This paper applied a feature selection to significantly improve time measurement in the three ML algorithms. Alongside the improvement of time measurement, this paper also compares the accuracy between using feature selection and without feature selection. The result of this comparison shows that higher accuracy is acquired by using full dataset, but in contrast, the time is much slower compared to the result of feature selection. Although this paper tried to show the impact of feature selection on time measurement, the detailed time measurement is not shown in a detailed unit.

Previous studies discussed the method of detecting fraud accounts by applying ML algorithm. However, these studies did not consider applying a DL algorithm yet, even though DL has the advantage of decreasing human intervention in the model. Also, most previous studies only show accuracy in evaluating their model performance. Hence, in this study, we use DL model

to classify the fraud account on the Ethereum blockchain and also tried to consider the detailed time measurement as an addition to model performance evaluation.

Ⅲ. Proposed System

The flowchart of the proposed system in this paper is shown in Fig. 1. First, the Ethereum dataset is split into two parts of the dataset; training dataset and testing dataset. The dataset training is used to train the DL model, and after the model is trained, the model is tested to classify either fraud or non-fraud based on the testing dataset. This section explains the DL model, dataset description, data preprocessing, and evaluation metrics.

3.1 CNN-based DL Model

The CNN-based model is used to accurately classify fraud accounts based on their transaction history. This paper also tried to compare the combination of CNN layers. The detailed

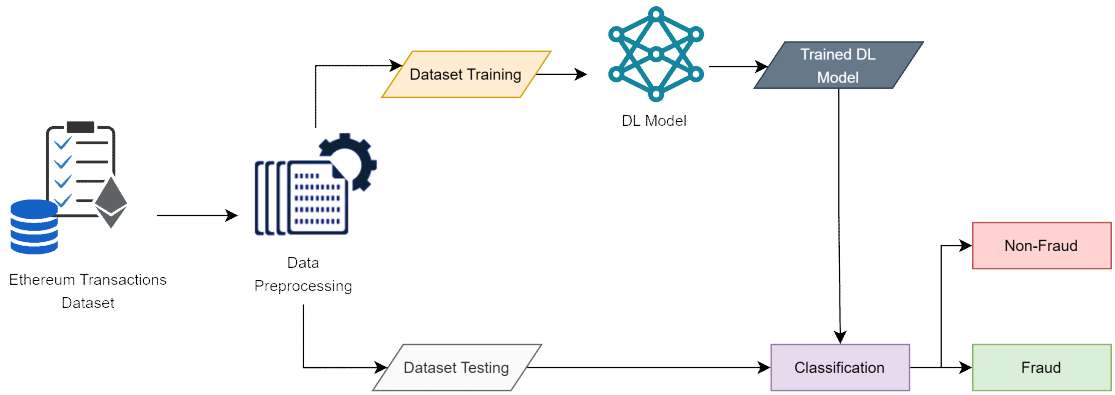

Fig. 1. Flowchart of the proposed system.

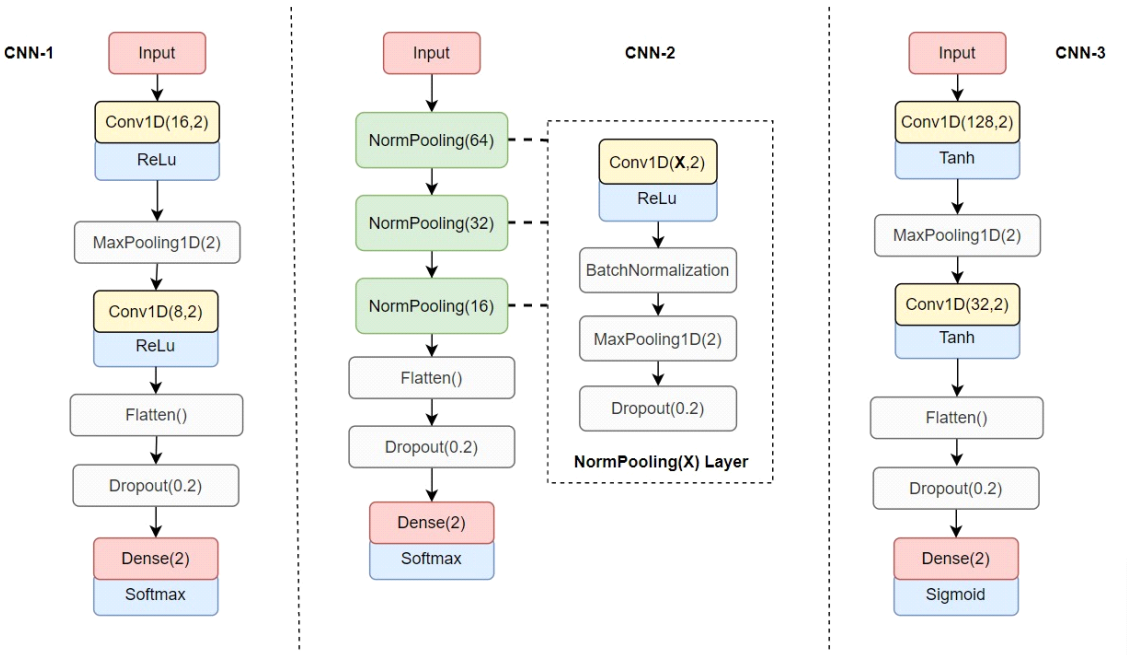

Fig. 2. CNN-based DL layer information.

configuration of each layer information is shown in Fig 2. Mainly, the maxpooling layer is used after each convolutional layer. The convolutional layer, in this case, Conv1D, is used to extract the transaction's features. The maxpooling layer reduces the number of features by applying the max operation along a sliding window in each feature dimension. In addition, maxpooling also could reduce overfitting and improve computational efficiency. Flatten is also used to reshape the result to a 1D vector from the previous convolutional layer so it can be fed into the output layer. Various numbers of neuron units are used in each block to reduce the number of trainable parameters as well as to minimize memory usage. In this paper, some activation functions will also be tried to be compared. These activation functions are:

1. ReLu: This activation function is used as it is computationally efficient, and it helps to solve the vanishing gradient problem.
2. Softmax: This activation function produces a probability distribution over the classes. Also, it is able to produce a probability distribution over the classes.
3. Tanh: Tanh is used to help to introduce nonlinearity in the network and make it more capable of modeling complex relationships between inputs and outputs. The output of Tanh is between -1 and 1, which could be useful in some cases.
4. Sigmoid: It is a mathematical function that maps any input value to a value between 0 and 1, making it useful for binary classification problems.

Table 1. The layer information for FNN and LSTM

FNN		LSTM	
Layer Information	Activation Function	Layer Information	Activation Function
Dense(64)	ReLu	LSTM(128)	ReLu
Dense(8)	ReLu	LSTM(64)	ReLu
Dropout(0.4)	-	Dense(32)	ReLu
Dense(2)	Softmax	Dropout(0.2)	-
		Dense(2)	Softmax

3.2 Other DL Model

We also tried to compare it with other DL models in this work. We use some basic models to compare the CNN method in this case. These models are Feedforward Neural Network (FNN) and Long-Short Term Memory (LSTM). The layers used for designing the basic FNN and LSTM are provided in Table 1. The activation layer in these layers is ReLu and softmax. ReLu is used for each activation function in each layer, and the softmax is used for the output layer.

3.3 Dataset Description

The dataset is obtained from [18], which contains fraud and valid transactions made over Ethereum. This dataset also used in [12] and [13] to test the machine learning model. This dataset contains 9841 transactions made over Ethereum. To be exact, this dataset contained an imbalanced dataset consisting of 7662 non-fraud transactions and 2179 fraud transactions. It has 46 features that consist of various detailed transactions from the average time taken for a transaction until the average value of the transaction.

Table 2. Dataset features layout example.

Avg min between sent tnx	Avg min between received tnx	Time diff between first and last (Mins)	Avg min between sent tnx	...	ERC20 avg val sent	ERC20 max val sent
844.26	1093.71	704785.63	721	...	271779.92	1.6831e+07
12709.07	2958.44	1218216.73	94	...	2.2608	2.260880e+00
246194.54	2434.02	516729.30	2	...	0.00	0.000e+00
...	...	...	...	...	...	...
...	...	...	...	...	...	...
2499.44	2189.29	261601.88	67	...	0.00	0.00e+00
37242.70	149.56	670817	18	...	0.00	0.00e+00

The layout of the dataset is shown in Table 2.

3.4 Preprocessing

The gap between non-fraud transactions and fraud transactions is big, so this dataset is considered an imbalanced dataset issue. Thus, to handle this problem, this paper implements a SMOTE Algorithm to create an artificial sample from the existing dataset. The ratio that is used from SMOTE is 70%, resulting in a new dataset containing 7662 non-fraud transactions and 5123 fraud transactions. The increasing fraud transactions dataset is done to handle the big gap ratio between the non-fraud and fraud classes. In addition, this paper also tried to reduce the unimportant features used from the dataset.

This paper reduces the dataset's features based on each feature's variance value. The feature that has zero variance in the dataset is removed, as this is because these features are considered as a not impactful feature model to learn the dataset. Based on the variance score, 13 features got removed. After selecting the features based on variance, feature scaling is done using normalization. Feature scaling is done so that the deep learning model converges faster. In addition, doing the feature scaling will help the activation function of *sigmoid*, *tanh*, and *softmax*, as these activation functions are sensitive to the input data. In addition, based on the correlation value for each feature, some features have a high correlation. These are three features: ERC20 total Ether sent, total Ether balance, and ERC20 total Ether sent a contract.

3.5 Evaluation Method

An accurate classification is needed to solve the fraud classification problem. Several performance indicators are calculated to evaluate the performance of the proposed model among all DL model comparisons, including:

1. Classification Accuracy. This metric calculates the proportion of instances that are correctly classified. Classification accuracy could be calculated using:

$$Accuracy = \frac{T_p + T_n}{T_p + T_n + F_p + F_n}, \tag{1}$$

with T_p, T_n, F_p, and F_n are true positive, true negative, false positive, and false negative, respectively.

2. Classification Loss. Classification loss measures the degree of an error made by a model in predicting the appropriate class for each instance, which is opposite to classification accuracy.
3. F1-Score. It is a measure of the harmonic mean between precision and recall. Precision is the proportion of true positives out of all predicted positives, while recall is the proportion of true positives out of all actual positives. F1-Score follow these equations:

$$Precision = \frac{T_p}{T_p + F_p}, \tag{2}$$

$$Recall = \frac{T_p}{T_p + F_n}, \tag{3}$$

$$F1 - Score = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)}. \tag{4}$$

4. Model Complexity. The time required to train the data, also known as the training time, and the time needed to identify an error from a single sample, also known as the inference time, are included in the model evaluation metrics.
5. Memory Usage. To determine how many operations are needed for a single forward pass, the floating point operation (FLOPs) is used to calculate the number of operations as the metrics to evaluate the memory usage

Ⅳ. Simulation Result

The system is evaluated on Google Colaboratory using the Python-based TensorFlow framework library for training the deep learning algorithm.

4.1 Classification Accuracy

Fig. 3 shows the accuracy of all DL models used in this work. The range of the accuracy score is from 0 to 1. The closer the accuracy value to 1, the better the model accuracy is. The model accuracy for each

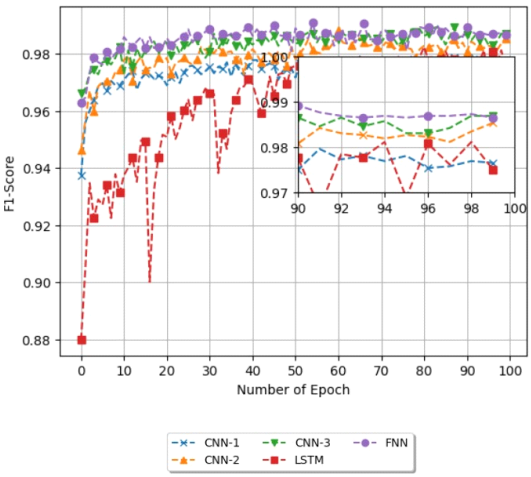

Fig. 3. The accuracy of the training and validation of the proposed model.

DL model can reach above 0.9. This behavior demonstrates that in terms of accuracy, the DL model could perform well in classifying the dataset. Based on Fig. 3, the CNN-3 model achieved the highest accuracy with an accuracy of 0.9867. Followed by FNN with a slightly different of 0.0004. The other model can also perform well, as shown by CNN-2 with 0.985 and CNN-1 with 0.976. Although LSTM is the least accurate, it is still considered a good result because it got an accuracy of 0.975.

4.2 Classification Loss

As previously mentioned, loss is one of the metrics to show the model's inaccuracy when the data is trained. This metric could give information on the effectiveness of each DL model that is used in this work. If the error rate of the loss is large, then the loss will also be high. This performance indicates that the model needs to learn more. On the other hand, the better the model, the smaller the loss that results from the model.

Fig. 4 shows the classification loss of each algorithm used in this work. In this figure, CNN-3 is able to achieve a low loss score of 0.0355 and followed by CNN-2 with a loss of 0.0398. Each model's loss value is close to 0; this indicates that the model could learn effectively. Loss-wise, the CNN-3 model is able to achieve the best result.

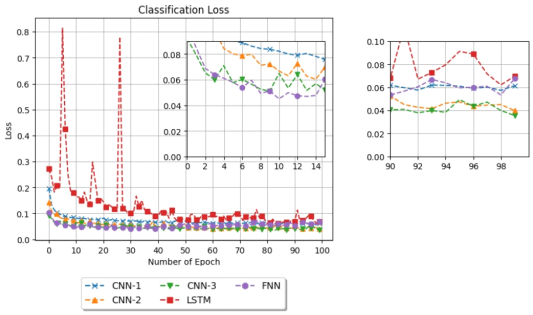

Fig. 4. The loss of training and validation loss of the proposed model.

4.3 F1-Score Comparsion

Another performance metric to validate the performance of DL or ML model is by using precision and recall. These metrics are used to evaluate the classification model' s prediction ability. As the precision increase, the recall value will decrease. This paper uses F1-score metrics to evaluate the mode, as the F1-score metrics is the harmonic between precision and recall metrics. F1-Score is ranged between 0 and 1. As the F1-score is closer to 1, it is a sign that the model performance is good. The F1-score comparison for all of the algorithms is shown in Fig. 5. Based on Fig. 5, all of the algorithms able to achieve an F1-score above 0.90 that indicates all the models are generally performing well. The highest F1-score is achieved by the CNN-3 model with 0.9751 compared to other DL models. In addition, all CNN models have an F1-Score better compared to LSTN and FNN The lowest F1-Score is achieved by LSTM with 0.9431.

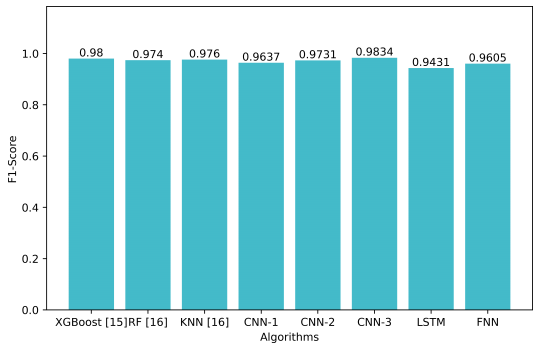

Fig. 5. F1-Score comparison of all algorihms.

4.4 Model Complexity

The model complexity comparison of each deep learning approach used in this paper is shown in Table 3. As shown in Table 3, the training and inference time is increased as the number of trainable parameters increases in the DL model. In this case, CNN-3 model is able to achieve the third-best inference time with a difference of 173 ms from the best inference time of CNN-1 model with a score of 204.29. LSTM achieves the longest inference time with an inference time of 1905 ms. But, compared to KNN model proposed by [16], the KNN model is slightly faster than CNN model in terms of inference time. CNN-1 model is also able to achieve the lowest training time and trainable parameter followed by FNN model. While LSTM is the highest trainable parameter with a number of trainable parameters of 118114 and a training time of 2964 seconds. Based on these parameters, CNN models are able to achieve a better result compared to other models, especially CNN-1 model.

4.5 Memory Usage

In addition to model complexity, this paper also adds memory usage as the performance metric of the DL model. Memory usage for a DL model is computed in floating point operations (FLOPs). FLOPs determines the number of operation for a single forward pass. The memory usage comparison of each DL model is shown in Table. 5. In this case, the CNN model has higher memory usage in comparison to LSTM and FNN. The lowest CNN model is achieved by CNN-1 model as it has the least neurons compared to other CNN models. The LSTM model achieved the least memory usage with a score

Table 5. Memory usage of all DL algorithms.

DL Model	FLOPs (MFLOPs)
CNN-1	0.203
CNN-2	2.8
CNN-3	4.93
LSTM	0.0708
FNN	0.092

Table 3. The model complexity comparison among all compared algorithms.

Model	Trainable Parameter	Training Time (s)	Inference Time (ms)
RF [16]	-	-	4850
KNN [16]	-	-	100
CNN-1	442	263.96	204.29
CNN-2	5682	504.06	387.83
CNN-3	9122	382.8	377.252
LSTM	118114	2964.9	1905.647
FNN	2906	202.79	251.31

Table 4. Impact of SMOTE in the performance metrics.

Model	Before SMOTE		After SMOTE	
	Precision (%)	Recall (%)	Precision (%)	Recall (%)
XGBoost [15]	-	-	97	98
RF [16]	-	-	97.5	98
KNN [16]	-	-	97.4	97.5
FNN	95.32	96.45	96.54	95.94
LSTM	96.36	94.08	95.10	94.10
CNN-1	94.12	94.79	96.63	96.29
CNN-2	97.36	95.97	97.50	97.26
CNN-3	96.36	94.08	98.41	98.32

of 0.0708.

4.6 SMOTE Impact Comparison

In this subsection, the impact of SMOTE algorithm is discussed. The result can be seen in Table 4. Overall in the Deep learning model. the impact of SMOTE algorithm is shown by the increase in recall. As the Recall increase, the trade-off with the precision happens. A high recall is considered a good thing in fraud detection cases as it is better to detect fraud in most cases. However, the precision also can not be left alone. Based on Table 4, the CNN-3 model is better than other models because it has a significant improvement compared to the result before SMOTE algorithm with a score of precision 98.41% and recall 98.32%. Overall, all the models achieved a good result with a score over 90% in weighted precision and recall after SMOTE Algorithm.

Ⅴ. Conclusion

A method for identifying malicious accounts in the Ethereum blockchain network based on the transaction history of the account was proposed in this paper. The features related to transactions performed by an account are used as indicators of malicious accounts. In this study, 9841 transactions were used for classification.

This study applied a deep learning model to identify fraudulent transactions. Various combinations of CNN and other DL models were also considered in this study. Based on the results, CNN was better than the other deep learning models regarding precision, recall, and F1-Score. Moreover, the CNN-3 model achieved a better result in terms of classification accuracy compared with other deep learning models. Based on the performance evaluation, we conclude that the CNN-3 model provides the best result compared with the other CNN models. This was indicated by the highest classification accuracy and F1-score of 0.9867 and 0.9834, respectively. The inference and training times were also low, 382.8 s and 377.252 ms, respectively. In future studies, we will implement this model using more datasets. Moreover, additional feature selection is required to select an effective feature from the dataset.

References

[1] A. Mashatan, M. S. Sangari, and M. Dehghani, "How perceptions of information privacy and security impact consumer trust in crypto-payment: An empirical study," *IEEE Access*, vol. 10, pp. 69 441-69 454, 2022. (https://doi.org/10.1109/ACCESS.2022.3186786)

[2] M. S. Nisha, G. Rajakumar, P. Jenifer, and B. Benita, "Protecting bitcoins frequency count against double-spend attacks," in *2023 5th ICSSIT*, pp. 725-731, 2023. (https://doi.org/10.1109/ICSSIT55814.2023.10060988)

[3] M. Hashemi Joo, Y. Nishikawa, and K. Dandapani, "Cryptocurrency, a successful application of blockchain technology," *Managerial Finance*, vol. 46, no. 6, pp. 715-733, 2023. (https://doi.org//10.1108/MF-09-2018-0451.)

[4] I. S. Igboanusi, K. P. Dirgantoro, J.-M. Lee, and D.-S. Kim, "Blockchain side implementation of pure wallet (pw): An offline transaction architecture," *ICT Express*, vol. 7, no. 3, pp. 327-334, 2021, ISSN: 2405-9595. (https://doi.org//10.1016/j.icte.2021.08.004) [Online] Available: https://www.sciencedirect.com/science/article/pii/S2405959521000928.

[5] Y. Chen and C. Bellavitis, "Blockchain disruption and decentralized finance: The rise of decentralized business models," *J. Business Venturing Insights*, vol. 13, no. e00151, 2020, ISSN: 2352-6734. (https://doi.org/10.1016/j.jbvi.2019.e00151) [Online] Available: https://www.sciencedirect.com/science/article/pii/S2352673419300824.

[6] H. Tran-Dang, S. Bhardwaj, T. Rahim, A. Musaddiq, and D.-S. Kim, "Reinforcement learning based resource management for fog computing environment: Literature review,

challenges, and open issues," *J. Commun. and Netw.*, vol. 24, no. 1, pp. 83-98, 2022. (https://doi.org/10.23919/JCN.2021.000041).

[7] B. E. Mykulyak, "Facilitating online cryptopayments now and in the future," *The PayTech Book: The Payment Technology Handbook for Investors, Entrepreneurs and FinTech Visionaries*, pp. 132-133, 2019. (https://doi.org/10.1002/9781119551973.ch40)

[8] S. Vyas, M. Gupta, and R. Yadav, "Converging blockchain and machine learning for healthcare," in *2019 AICAI*, pp. 709-711, 2019. (https://doi.org/10.1109/AICAI.2019.8701230)

[9] M. J. J. Gul, A. Paul, S. Rho, and M. Kim, "Blockchain based healthcare system with artificial intelligence," in *2020 Int. Conf. CSCI*, pp. 740-741, 2020. (https://doi.org/10.1109/CSCI51800.2020.00138)

[10] R. Akter, M. Golam, V.-S. Doan, J.-M. Lee, and D.-S. Kim, "Iomt-net: Blockchain integrated unauthorized UAV localization using lightweight convolution neural network for internet of military things," *IEEE Internet of Things J.*, pp. 1-1, 2022. (https://doi.org/10.1109/JIOT.2022.3176310)

[11] M. Golam, R. Akter, R. Naufal, V.-S. Doan, J.-M. Lee, and D.-S. Kim, "Blockchain inspired intruder UAV localization using lightweight cnn for internet of battlefield things," in *MILCOM 2022*, pp. 342-349, 2022. (https://doi.org/10.1109/MILCOM55135.2022.10017795).

[12] M. R. R. Ansori, Allwinnaldo, R. N. Alief, I. S. Igboanusi, J. M. Lee, and D.-S. Kim, "Hades: Hash-based audio copy detection system for copyright protection in decentralized music sharing," *IEEE Trans. Netw. and Serv. Manag.*, pp. 1-1, 2023. (https://doi.org/10.1109/TNSM.2023.3241610)

[13] M. R. R. Ansori, Allwinnaldo, R. N. Alief, I. S. Igboanusi, J. M. Lee, and D.-S. Kim, "Duplicate audio data detection model based on ipfs and blockchain using smart contract," in *Proc. KICS Winter 2022*, pp. 1-1, 2022.

[14] D. Kim, I. Doh, and K. Chae, "Improved raft algorithm exploiting federated learning for private blockchain performance enhancement," in *2021 ICOIN*, pp. 8228-832, 2021. (https://doi.org/10.1109/ICOIN50884.2021.9333932)

[15] A. Maurya and A. Kumar, "Credit card fraud detection system using machine learning technique," in *2022 IEEE Int. Conf. CyberneticsCom*, pp. 500-504, 2022. (https://doi.org/10.1109/CyberneticsCom55287.2022.9865466)

[16] R. F. Ibrahim, A. Mohammad Elian, and M. Ababneh, "Illicit account detection in the ethereum blockchain using machine learning," in *2021 ICIT*, pp. 488-493, 2021. (https://doi.org/10.1109/ICIT52682.2021.9491653)

[17] M. Ostapowicz and K. Żbikowski, "Detecting fraudulent accounts on blockchain: A supervised approach," in *WISE 2019*, R. Cheng, N. Mamoulis, Y. Sun, and X. Huang, Eds., Cham: Springer International Publishing, pp. 18-31, 2019, ISBN: 978-3-030-34223-4.

[18] V. Aliyev, *Ethereum fraud detection dataset*, 2021. [Online] Available: https://www.kaggle.com/datasets/vagifa/ethereum-frauddetection-dataset.

Revin Naufal Alief

He earned his Bachelor degree in Telecommunication Engineering from Telkom University, Indonesia in 2020. He is currently pursuing master's degree in IT Convergence Engineering in the: Kumoh National Institute of Technology, Gumi, South Korea. He is also a researcher in Network System Laboratory, one of the laboratories at Kumoh National Institute of Technology. His re- search interest is regarding blockchain, information and security, and artificial intelligence.

Syifa Maliah Rachmawati

She received her B.S. degree from Telkom University, Indonesia in 2018. From 2019 to 2021, she worked as Software Developer in Indonesia. Currently, she works as a full-time researcher and pursue her M.S. degree in the Department of IT Convergence Engineering, Kumoh National Institute of Technology, South Korea. Her research in- terests are in the area of deep-learning techniques and the application of machine learning algorithms in the industrial internet of things.

Jae Min Lee

Jae-Min Lee received the Ph. D degree in electrical and computer engineering form the Seoul Na- tional University, Seoul, Korea, in 2005. From 2005 to 2014, he was a Senior Engineer with Samsung Electronics, Suwon, Korea. From 2015 to 2016, he was a Principle Engineer in Samsung Electronics, Su- won, Korea. Since 2017, he has been an associate professor with School of Electronic Engineering and Department of IT-Convergence Engineering, Kumoh National Institute of Technology, Gyeongbuk, Korea. He is a member of IEEE. His current main research interests are smart IoT convergence application, in- dustrial wireless control network, UAV, Metaverse and Blockchain.

[ORCID: 0000-0001-6885-5185]

Dong-Seong Kim

2003 : Ph.D. Electrical and Computer Engineering, Seoul National University, Korea.

2003~2004 : Postdoctoral resea- rcher, Cornell University, NY, USA

2007~2009 : Visiting Professor, The University of California, Davis, CA, USA

2004~Current : Professor, Kumoh National Institute of Technology (KIT), Gyeongbuk, Korea

2014~Current : Director, ICT Convergence Research Center, KIT, Gyeongbuk, Korea

2017~2022 : Dean, Industry-Academic Cooperation Foundation and Office of Research (ICT), KIT, Gyeongbuk, Korea

2022~Current : CEO, NSLab Co. Ltd., Korea

<Research Interests> Blockchain, Metaverse, Industrial IoT, real-time systems, industrial wireless control network, 5G+, and 6G.

[ORCID:0000-0002-2977-5964]