DDoS Attack Detection on Bitcoin Ecosystem using Deep-Learning

Ui-Jun Baek
Computer Information and Science
Korea University
Sejong, Korea
pb1069@korea.ac.kr

Se-Hyun Ji
Computer Information and Science
Korea University
Sejong, Korea
sxzer@korea.ac.kr

Jee Tae Park
Computer Information and Science
Korea University
Sejong, Korea
pjj5846@korea.ac.kr

Min-Seob Lee
Computer Information and Science
Korea University
Sejong, Korea
chenlima2@korea.ac.kr

Jun-Sang Park
A&B Center
LG Electronics
Seoul, Korea
junsang.park@lge.com

Myung-Sup Kim
Computer Information and Science
Korea University
Sejong, Korea
tmskim@korea.ac.kr

***Abstract*—Since Bitcoin, the first cryptocurrency that applied blockchain technology was developed by Satoshi Nakamoto, the cryptocurrency market has grown rapidly. Along with this growth, many vulnerabilities and attacks are threatening the Bitcoin ecosystem, which is not only at the bitcoin network-level but also at the service level that applied it, according to the survey. We intend to analyze and detect DDoS attacks on the premise that bitcoin's network-level data and service-level DDoS attacks with bitcoin are associated. We evaluate the results of the experiment according to the proposed metrics, resulting in an association between network-level data and service-level DDoS attacks of bitcoin. In conclusion, we suggest the possibility that the proposed method could be applied to other blockchain systems.**

Keywords—Bitcoin, DDoS, Detection, Deep-Learning

I. INTRODUCTION

A. Backgrounds

Bitcoin is the first cryptocurrency which is based on block-chain technology and has been widely adopted[1]. This cryptocurrency challenged the currency market as a clean alternative that guarantees anonymity and is not under the control of the central governments. Despite the uncertainty of whether the bitcoin is a suitable alternative to existing currencies[2-3], Bitcoin still remains the highest trading volume among all cryptocurrency[4]. With the growth of the bitcoin market, many vulnerabilities and attacks have been investigated[5-6], and these vulnerabilities and attacks have been found to affect the bitcoin eco system[7]. In addition to the technical vulnerabilities of bitcoin, attacks at related service levels can also affect the bitcoin ecosystem. One of the representative examples is the collapse of Mt.gox, which was the largest exchange[8]. The attacks that occur at these service levels occupy a large number of DDoS attacks, and studies for analyzing such DDoS attacks have been conducted[9]. But, there is no study for detecting DDoS attacks in practical.

B. Proposed Method

This paper focuses on the detecting DDoS attacks on service level including mining pool and exchange. The outline of the proposed method is as follows:

1. Collecting of DDoS attack data and Bitcoin network data(Blocks, Transactions)
2. Pre-processing of collected data (1)
3. Extracting statistical data from (2)
4. Feature extraction from (3)
5. Training, Validating, Testing
6. Evaluating

Following the introduction of Section I, Section II lists and classifies related works. Section III describes the pre-processing method and the deep-learning method, and Section IV describes the experimental procedure including data pre-processing. Section V evaluates the experimental results according to the proposed evaluation metrics. Section VI describes the conclusions, including limitations and future works

II. RELATED WORKS

A. Analysis of DDoS on Bitcoin Network

In general, the Bitcoin network is known to be robust against DDoS attacks due to the characteristic of distributed ledgers. But, bitcoin network still can potentially be attacked by DDoS attacks, and possible attack is message spoofing using Transmission Control Protocol (TCP) sequence numbers[10]. There are also many DDoS attacks at the service level where bitcoin is being used. The DDoS attacks at the service level cannot directly affect the performance of the network or steal the currency, but they are closely related to the value of the currency and eventually lead to the BTC's depreciation[8, 11, 12]. Most services related to bitcoin where DDoS attacks occur are on the exchanges or the mining pools[9]. The short-term and long-term effects have been analyzed through game theory models when conducting DDoS attacks between competing for mining pools [13].

B. Anomaly Detection for Bitcoin Network

Studies have been conducted to detect anomaly in Bitcoin network using machine learning. [14] used two types of graphs to analyze the behavior patterns of users and transactions, and they detected three of the 30 known cases They attained similar results using k-means clustering, Local Outlier Factor(LOF) and power degree & densification in their subsequent study[15]. [16] attempted to distinguish between normal users and malicious users based on real

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea Government(MSIT) (No.2018-0-00539-001, Development of Blockchain Transaction Monitoring and Analysis Technology) and Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2018R1D1A1B07045742)

reported cases. [17] used trimmed k-means clustering for anomaly detection.

C. Analysis for Bitcoin Network using Deep-Learning

Bitcoin network analysis based on Deep-Learning mainly consists of market price prediction of cryptocurrency and user classification. [18] used autoregressive integrated moving average (ARIMA) model, recurrent neural network(RNN), long short term memory(LSTM) for the market price prediction of BTC. [19] presented an approach for applying LSTM directly to such graph neighborhoods, yielding predictions for graph nodes on the basis of the structure of their local neighborhood and the features of the nodes in it. [20] proposed a deep learning method to achieve address-user mapping.

III. KEY CONCEPTS

A. Principal Component Analysis(PCA)

PCA is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables (entities each of which takes on various numerical values) into a set of values of linearly uncorrelated variables called PC(principal component)s[21]. The brief is as follows:

Assuming that the average of the dataset is zero, the principal component w_1 of the dataset x is defined as:

$$w_1 = \arg\max_{\|w\|=1} E\ \{(w^T x)^2\}$$

When k-1 PCs are already given, the kth PC can be found by subtracting the previous k-1 PCs:

$$\hat{x}_k = x - \sum_{i=1}^{k-1} w_i w_i^T\, x$$

Then subtract this value from the dataset and find the new PC:

$$w_k = \arg\max_{\|w\|=1} E\ \{(w^T \hat{x})^2\}$$

B. Multi-Layer Perceptron

MLP(Multi-Layer Perceptron) is a class of feedforward artificial neural network and consists of at least three layers of nodes[22]. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. The two common activation functions are both sigmoids and where y_i is the output of the ith node and v_i is the weighted sum of the input connections, the formulas of sigmoids are as follows:

$$y(v_i) = \tanh(v_i), y(v_i) = (1 + e^{-vi})^{-1}$$

The learning process is changing connection weights based on the amount of error in the output compared to the expected result. Where d is the target value and y is the value produced by the perceptron, error in output node j in the nth data point is as follows:

$$e_j(n) = d_j(n) - y_j(n)$$

The weights of nodes are adjusted based on corrections that minimize the error in the entire output and the error of the data n is as follows:

$$\varepsilon(n) = \frac{1}{2}\sum_j y_i(n)$$

Where y_i is the output of the previous neuron and γ is the learning rate, the change in each weight of nodes is as follows:

$$\Delta w_{ji}(n) = -\gamma \frac{\partial \varepsilon(n)}{\partial v_j(n)} y_i(n)$$

MLP utilizes a supervised learning technique called backpropagation for training. After adjusting the weight of all nodes to the number of epoch fixed, learning is finished and the output of its are weights of nodes as a result.

IV. EXPERIMENT

A. Collecting Data

The data we collect and analyze is real cases of the DDoS attacks on services related to Bitcoin. The real case data of the DDoS attack is the data used in previous DDoS attack analysis study[9] and was downloaded from [23] and is based on reports of Bitcoin forum site[24]. The DDoS attack data contain the name of service which is attacked by DDoS, the date that DDoS attack conducted, the category of service and the number of posts which report DDoS attack. DDoS attack data are reported cases from May 2011 to October 2013, with detailed information on attacks shown in Table 1. Then we collect block data generated at the time of the DDoS attack. The block data can be collected through the *getblock* command of Bitcoin client program, and transaction data contained in the block was also collected by setting *verbosity* option to 2.

TABLE I. DETAIL OF DDOS ATTACK DATA

Category	Number of reports	Proportion
Mining pool	54	38%
Currency exchange	58	41%
e-Wallet	6	4%
Financial	7	5%
Gambling	13	9%
Other	4	3%

B. Extracting Statistical Data

We extract the statistical data from the collected Bitcoin block data and Bitcoin transaction data. We extracted statistical data such as summation, maximum, minimum, average, standard variation. The extraction criteria are based on block. The statistical data extracted are described in table 2. Each raw data went through 0 to 2 statistical extracting

TABLE II. EXTRACTED STATISTICAL DATA

Data Level	Raw data (0^{th})	1st Extraction	2nd Extraction	Number of data
Block	nTx			1
	Weight			1
	Size			1
	vSize			1
Transaction	nVin		Sum Max Min Avg Stdv (5)	5
	nVout			5
	Value			5
	Fee			5
	Tx_vSize			5
	Tx_Size			5
Input & Output of Transaction	Vout_value	Sum Max Min Avg Stdv (5)		25
	Vin_value			25

process depending on its data level, resulting in a total of 84 data extracted.

C. Pre-Processing

We use PCA to perform feature extraction to eliminate redundant and unnecessary data from the extracted data. Feature extraction using PCA was performed until at least 99% of the dimension of the original data express.

D. Detecting DDoS using Deep-Learning

We use MLP to detect DDoS attack data and all data is divided into a training set, validation set, and testing set with a division ratio of 6:2:2. The input parameters include the number of hidden layers, the learning rate, the number of nodes in each layer, and the number of learning epochs. we fix the learning rate to 0.01 and fix the number of nodes in each hidden layer to the size of the input layer. For various experiment result, we change only the number of hidden layers and the learning epochs. All data is labeled as DDoS or normal, respectively, and the criterion to be labeled is whether the block was created on the day the DDoS attack occurred.

V. RESULTS

A. Result of the Traning Set

We show the experimental results on various parameters in 3d-graphs and analyzed them through this graph. Fig. 1 - A shows the model accuracy of detecting DDoS attacks in the training set. The model shows high accuracy except for too little learning or too much learning. In addition, the higher the number of hidden layers, the higher the number of hidden layers, the higher the accuracy of the model. Fig. 2 - B shows the model accuracy of detecting Non-DDoS attacks in the training set. The model has high detection accuracy, which shows that the model has been well trained. The results in Fig. 1 show that the block created when the DDoS attack occurred is characteristically distinct from the block created when a DDoS attack did not occur.

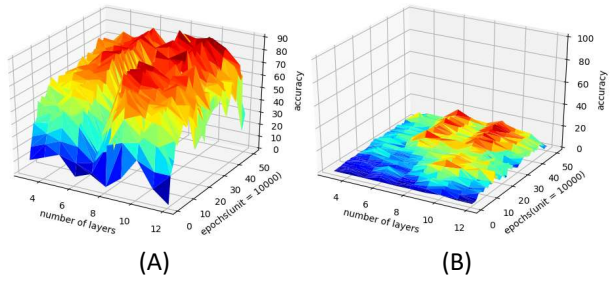

Figure 1. Accuracy of the DDoS Attack detection(A) and the normal block detection(B) in training set

B. Result of the Validation Set

We analyze the results of the validation set to select the model with the optimal hyper-parameters. Fig. 2-A shows the model accuracy of detecting DDoS attacks in the validation set. The model shows high accuracy when the number of hidden layers was set at 9 or 11. Fig. 2-B shows the model accuracy of detecting Non-DDoS attacks in the validation set and shows the same result as the result in the training set. We check the experimental results in the test set with the model in the top 5 showing high results in the validation set. These experimental results are shown in Table 3.

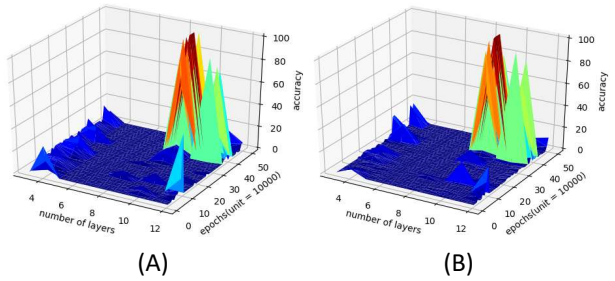

Figure 1. Accuracy of the DDoS Attack detection(A) and the normal block detection(B) in validation set

TABLE III. TOP 10 HIGH ACCURACY IN VALIDATION SET

#Layers	#Epochs	Validation (DDoS)	Validation (Non-DDoS)	Test (DDoS)	Test (Non-DDoS)
9	440000	100	100	0	0
9	450000	100	100	0	0
9	460000	100	100	0	0
9	430000	100	100	0	0
9	390000	100	100	0	0

In Table 3, we can see that the model shows high accuracy in the validation set but is not detectable at all in the test set. We analyzed the reasons for the above phenomenon through the results of the experiment on the test set.

C. Result of the Test Set

Fig. 3 shows the model accuracy of detecting DDoS attacks and Non-DDoS attacks in the test set, respectively. Fig. 3 shows that the model has high accuracy with 3 or 12 hidden layers. But, as shown in Fig. 1 - B, training is not good when the number of hidden layers is 3. Thus the model shows high accuracy with 12 layers and higher training epochs. We check the experimental results in the validation set with the model in the top 5 showing high results in the test set. These experimental results are shown in Table 4. The models with high accuracy at the test set shows low accuracy at the validation set. We determine through these extreme results that data of test set and data of validation set have different data patterns that are mutually exclusive.

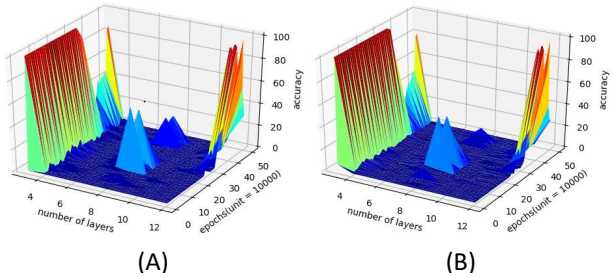

Figure 3. Accuracy of the DDoS Attack detection(A) and the normal block detection(B) in test set

TABLE IV. TOP 10 HIGH ACCURACY IN TEST SET

#Layers	#Epochs	Test (DDoS)	Test (Non-DDoS)	Validation (DDoS)	Validation (Non-DDoS)
12	420000	100	100	2.94	4.73
12	390000	100	100	5.88	0.68
12	410000	100	100	2.94	0.68
12	470000	100	100	0	0
12	380000	100	96.15	2.94	4.73

D. General Performance of Proposed Method

We determined from the above results that there are two mutually exclusive patterns in the DDoS data set. Thus, multiple models for both patterns can ensure a high accuracy. However, we conducted experiments on randomly shuffled data set to draw general detection accuracy for a single model we proposed. We keep the training set intact and only shuffle validation set and test set to measure the performance of a single model when random data set is entered into the model. n this experiment, the unit of epoch is set to 100 for more detailed experimentation. Performance measurements from shuffled data are shown in the Table 7, 8. Table 7, 8 shows the performance of single model with shuffled data. Overall, the accuracy of detecting DDoS attack is about 50%, and the accuracy of distinguishing normal block data about 70%. As a result, it can be said that the single model has poor results and that it is appropriate to use multiple models to actually use the method proposed in this paper.

TABLE V. PERFORMANCE WITH SHUFFLED DATA (HIGH ACCURACY ON VALIDATION SET)

#Layers	#Epochs	Validation (DDoS)	Validation (Non-DDoS)	Test (DDoS)	Test (Non-DDoS)
12	286700	55.46	76.72	53.17	68.79
12	470700	51.28	68.15	50.07	67.16
12	470800	51.28	68.15	50.07	67.16
12	470900	51.28	68.15	50.07	67.16
12	471000	51.28	68.15	50.07	67.16

TABLE VI. PERFORMANCE WITH SHUFFLED DATA (HIGH ACCURACY ON TEST SET)

#Layers	#Epochs	Validation (DDoS)	Validation (Non-DDoS)	Test (DDoS)	Test (Non-DDoS)
12	395500	54.49	73.51	55.12	72.36
12	396000	54.38	64.27	54.89	69.85
12	394500	54.38	64.27	54.89	69.85
12	394800	54.38	64.27	54.89	69.85
12	395400	52.58	64.27	54.89	69.85

VI. CONCLUSION

In this paper, we proposed using bitcoin data to predict the DDoS attack that has occurred in services related to bitcoin-related services. The proposed method defined the data that could be collected from the Bitcoin network and the statistical data of blocks that could be extracted from the collected data. We conducted detection experiments on various parameters and as a result, we derived the optimal number of hidden layers and the optimal number of train epochs. We analyzed the experimental results from each dataset and observed that our models did not cover both validation set and test set. but, we determined that this is due to the mutually exclusive characteristic of the data set and suggested a solution. In conclusion, we proposed a practical method of detecting DDoS attack on bitcoin-related service. The limitation of the proposed method is that it cannot detect DDoS attack by reflecting all the characteristics of the blocks created when DDoS attack occurs and that it is difficult to analyze how the characteristics of the blocks differ from those of normal blocks when DDoS attack occurs. Therefore, in future works, we will study how to extract features that can reflect all the characteristics of blocks created when DDoS attack occurs, and we will study specifically which features differ from normal blocks through deep analysis of data sets.

REFERENCES

[1] NAKAMOTO, Satoshi, et al. Bitcoin: A peer-to-peer electronic cash system. 2008.

[2] "Top 100 Coins by Market Capitalization." CoinMarketCap, last modified May 07. 2019, accessed Apr 30. 2019, https://coinmarketcap.com/ko/coins/views/market-cap-by-total-supply/

[3] YERMACK, David. Is Bitcoin a real currency? An economic appraisal. In: Handbook of digital currency. Academic Press, 2015. p. 31-43.

[4] GLASER, Florian, et al. Bitcoin-asset or currency? revealing users' hidden intentions. Revealing Users' Hidden Intentions (April 15, 2014). ECIS, 2014.

[5] KIRAN, Mariam; STANETT, M. Bitcoin risk analysis. NEMODE Policy Paper, 2015.

[6] HASANOVA, Huru, et al. A survey on blockchain cybersecurity vulnerabilities and possible countermeasures. International Journal of Network Management, 2019, 29.2: e2060.

[7] BAQER, Khaled, et al. Stressing out: Bitcoin "stress testing". In: International Conference on Financial Cryptography and Data Security. Springer, Berlin, Heidelberg, 2016. p. 3-18.

[8] FEDER, Amir, et al. The impact of DDoS and other security shocks on Bitcoin currency exchanges: Evidence from Mt. Gox. Journal of Cybersecurity, 2018, 3.2: 137-144.

[9] VASEK, Marie; THORNTON, Micah; MOORE, Tyler. Empirical analysis of denial-of-service attacks in the Bitcoin ecosystem. In: International conference on financial cryptography and data security. Springer, Berlin, Heidelberg, 2014. p. 57-71.

[10] TAPSELL, James; AKRAM, Raja Naeem; MARKANTONAKIS, Konstantinos. An evaluation of the security of the Bitcoin Peer-to-Peer Network. arXiv preprint arXiv:1805.10259, 2018.

[11] GANDAL, Neil, et al. Price manipulation in the Bitcoin ecosystem. Journal of Monetary Economics, 2018, 95: 86-96.

[12] MARELLA, Venkata, et al. Bitcoin: A Social Movement Under Attack. Scandinavian IRIS Association, 2017.

[13] LASZKA, Aron; JOHNSON, Benjamin; GROSSKLAGS, Jens. When bitcoin mining pools run dry. In: International Conference on Financial Cryptography and Data Security. Springer, Berlin, Heidelberg, 2015. p. 63-77.

[14] PHAM, Thai; LEE, Steven. Anomaly detection in bitcoin network using unsupervised learning methods. arXiv preprint arXiv:1611.03941, 2016.

[15] PHAM, Thai; LEE, Steven. Anomaly detection in the bitcoin system-a network perspective. arXiv preprint arXiv:1611.03942, 2016.

[16] ZAMBRE, Deepak; SHAH, Ajey. Analysis of Bitcoin network dataset for fraud. Unpublished Report, 2013.

[17] MONAMO, Patrick; MARIVATE, Vukosi; TWALA, Bheki. Unsupervised learning for robust Bitcoin fraud detection. In: 2016 Information Security for South Africa (ISSA). IEEE, 2016. p. 129-134.

[18] MCNALLY, Sean; ROCHE, Jason; CATON, Simon. Predicting the price of Bitcoin using Machine Learning. In: 2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP). IEEE, 2018. p. 339-343.

[19] AGRAWAL, Rakshit; DE ALFARO, Luca; POLYCHRONOPOULOS, Vassilis. Learning From Graph Neighborhoods Using LSTMs. In: Workshops at the Thirty-First AAAI Conference on Artificial Intelligence. 2017.

[20] SHAO, Wei, et al. Identifying Bitcoin Users Using Deep Neural Network. In: International Conference on Algorithms and Architectures for Parallel Processing. Springer, Cham, 2018. p. 178-192.

[21] "Principal component analysis" Wikipedia, last modified May 04. 2019, accessed Apr 30. 2019, https://en.wikipedia.org/wiki/Principal_component_analysis

[22] "Multilayer perceptron" Wikipedia, last modifed May 05. 2019, accessed Apr 30, 2019, https://en.wikipedia.org/wiki/Multilayer_perceptron

[23] Vasek, Marie; Thornton, Micah; Moore, Tyler, 2014, "Replication data for: Empirical Analysis of Denial-of-Service Attacks in the Bitcoin Ecosystem", https://doi.org/10.7910/DVN/25541, Harvard Dataverse, V2

[24] "Bitcoin Forum" Bitcoin Forum, last modified May 07. 2019, https://bitcointalk.org/