Journal of Multimedia Information System VOL. 6, NO. 4, December 2019 (pp. 165-172): ISSN 2383-7632 (Online)
http://doi.org/10.33851/JMIS.2019.6.4.165

An Intrusion Detection Model based on a Convolutional Neural Network

Jiyeon Kim[1], Yulim Shin[2], Eunjung Choi[2*]

Abstract

Machine-learning techniques have been actively employed to information security in recent years. Traditional rule-based security solutions are vulnerable to advanced attacks due to unpredictable behaviors and unknown vulnerabilities. By employing ML techniques, we are able to develop intrusion detection systems (IDS) based on anomaly detection instead of misuse detection. Moreover, threshold issues in anomaly detection can also be resolved through machine-learning. There are very few datasets for network intrusion detection compared to datasets for malicious code. KDD CUP 99 (KDD) is the most widely used dataset for the evaluation of IDS. Numerous studies on ML-based IDS have been using KDD or the upgraded versions of KDD. In this work, we develop an IDS model using CSE-CIC-IDS 2018, a dataset containing the most up-to-date common network attacks. We employ deep-learning techniques and develop a convolutional neural network (CNN) model for CSE-CIC-IDS 2018. We then evaluate its performance comparing with a recurrent neural network (RNN) model. Our experimental results show that the performance of our CNN model is higher than that of the RNN model when applied to CSE-CIC-IDS 2018 dataset. Furthermore, we suggest a way of improving the performance of our model.

Key Words: Intrusion detection, Deep learning, Convolutional neural network, Recurrent neural network.

I. INTRODUCTION

Traditional rule-based security solutions hardly detect advanced attacks such as zero-day attacks and advanced persistent threats (APT). Attackers acquire advanced skills and exploit unknown vulnerabilities to bypass security solutions. Machine-learning (ML) is becoming a prevalent way of detecting advanced attacks with unexpected patterns [1]. ML is based on statistical and mathematical algorithms rather than rule-based algorithms. ML techniques contributes to improving performance of intrusion detection systems (IDS). Numerous studies have been addressing ML-based IDS techniques since KDD CUP 99(KDD) appeared in 1999. KDD is the most widely used dataset generated by Defense Advanced Research Projects Agency (DARPA) for IDS evaluation. KDD consists of four types of attack-labeled data including denial of service (DoS), probe, U2R (User-to-Root), and R2L (Remote-to-Local). Hasan [2] focuses on 2-class classification and multi-class classification using support vector machine (SVM). Mulay [3] employs Random Forest as well as SVM for intrusion detection and preprocess the KDD dataset through binary encoding and data rescale. Beghdad [4] classifies normal and malicious traffic based on SVM and then detect attacks based on Decision Tree.

Numerous studies employ deep-learning (DL) for intrusion detects. Jia [5] and Yuchen [6] proposes a convolutional neural network (CNN) model for IDS and trains KDD with images. Le [7], Staudemeyer [8], and Kim [9] suggest an IDS model based on LSTM-RNN. Later, NSL-KDD and gureKDD appeared to improve the problems of KDD. The latest well-known datasets are CIC IDS 2017(CIC-2017) [10] and CSE-CIC-IDS 2018 (CIC-2018) [11]. CIC-2017 contains recent attacks with similar form of PCAP. It was created based on B-Profile system [12]. After CIC-2017 released, several studies suggest intrusion detection model using CIC-2017 based on ML [13-14]. CIC-2018 is the most up-to-date dataset including common attacks for IDS evaluation. CIC-2018 was not generated based on KDD and consists of 7 types of attack scenarios-labeled data (specifically 16 types of attacks) including brute-force, DoS, and Botnet. CIC-2018 contains massive network traffic and system logs. We can hardly find CIC-2018 studies using DL compared to ML-based studies on CIC-2018 [15-16].

In this paper, we develop an intrusion detection model based on CNN, one of DL algorithms used to train image

Manuscript received November 27, 2019; Revised December 13, 2019; Accepted December 16, 2019. (ID No. JMIS-19M-11-037)
Corresponding Author (*): Eunjung Choi, 621 Hwarang-ro, Nowon-gu, Seoul, Tel +82-970-5339, E-mail. chej@swu.ac.kr
[1]Center for Software Educational Innovation, Seoul Women's University, Seoul, South Korea, jykim07@swu.a.kr
[2]Dept. of Information Security, Seoul Women's University, Seoul, South Korea, {yulim, chej}@swu.ac.kr

datasets. We first convert the CIC-2018 numerical data into images. We then develop a CNN-based intrusion detection model by organizing convolutional layers and max-pooling layers. Furthermore, we train the images based on the proposed model and evaluate its performance by comparing experimental results with that of a recurrent neural network (RNN) model. Lastly, we discuss on a way of improving the performance. CNN and RNN are fundamental deep learning models for image data and time-series data, respectively. Inception [25] as well as ResNet [26] are based on CNN. Long Short-Term Memory (LSTM) [27] is an advanced model of RNN. By employing these fundamental models, we are able to identify the optimal analysis model for the characteristics of CIC-2018. Furthermore, we could improve the performance using those advanced models in the future. The remainder of this paper is organized as follows. Section 2 briefly describes existing ML-based studies on intrusion detection as well as DL algorithms we use in this work. In Section 3, we design our CNN-based intrusion model along with features. We evaluate the proposed model discuss a preprocessing issue for the better performance in Section 4. Finally, the conclusion is in Section 5.

II. RELATED WORKS

KDD CUP 99(KDD) was generated for IDS evaluation and includes four types of attacks such as DoS, R2L, U2R, and probing. KDD consists of 41 features including traffic features, basic and content features of each TCP connection. KDD has been widely used for data mining and ML studies on intrusion detection. Table 1 shows existing ML/DL-based studies on intrusion detection using KDD.

Table 1. ML/DL-based intrusion detection studies using KDD CUP 99 and NSL-KDD.

Algorithm \ Dataset		KDD CUP 99									NSL-KDD
ML	SVM	O	O								
	ANN	O									
	Decision Tree	O									
DL	DNN					O					
	CNN		O								
	RNN			O			O				O
	LSTM			O	O		O				
preprocessing					O			O	O	O	
Reference		[17]	[6]	[8]	[9]	[18]	[7]	[19]	[20]	[15]	[21]

Some studies employ ML technique such as SVM, Decision Tree, and Artificial Neural Network (ANN) [6, 17]. Most of DL-based studies use CNN, RNN, LSTM and Deep Neural Network (DNN) algorithms [7-9], [17-18]. Moreover, some studies focus on preprocessing techniques of KDD [19-20]. NSL-KDD was generated to resolve some issues in KDD, especially duplicated records and lack of patterns of several attacks. Chuanlong [21] studies an intrusion detection model using Recurrent Neural Network (RNN) using NSL-KDD.

Canadian Institute for Cybersecurity (CIC) generated IDS datasets in 2012, 2017 and 2018. In 2012, ISCX IDS 2012 (CIC-2012)[22] was generated by injecting 4 types of attacks including infiltration attacks from inside, HTTP DoS attacks, DDoS(distributed denial of service) attacks and brute force attacks. Tamim [23] detects attacks in CIC-2012 based on CNN. He generates input images by converting destination payloads and classifies the images into normal and attack, while we classify two or more attacks in CIC-2018 based on a multi-class classification. CIC-2017 [10] and CIC-2018 [11] are the most up-to-date datasets for IDS evaluation. CIC-2017 contains network traffic with most common attack families including brute force attacks, heartbleed attacks, botnets, DDOS attacks and web attacks. Faker [13] studies intrusion detection using CIC-2017 and UNSW-NB15 datasets. This study removes socket information to prevent model overfitting. To reduce data size, they remove null values and unimportant traffic information. They also convert string values into numerical values and normalize the values. If there are missing data or infinite data, they make two versions of data set. First, replace all of missing and infinite data into average data. Second, remove all the missing and infinite data. They evaluate their model with the two kinds of datasets. As training algorithms, DNN (Deep Neural Network), Random Forest, and Gradient Boosting Tree classification are used. X. Zhang [14] focuses on intrusion detection using Deep Forest. They preprocess the datasets using based on the P-ZigZag encoding method and apply an inverse discrete cosine transform (IDCT) into the preprocessed datasets.

CIC-2018 contains more recent network traffic with/without attacks. CIC-2018 was generated by collecting network traffic and system logs for about 80 features. Qianru [15] analyzes the CIC-2018 dataset employing ML techniques. This study preprocesses the dataset by eliminating normal data and noise data, and then remove unnecessary values after decimal point. With these preprocessing methods, the size of CIC-2018 decreased by 4MB.

Table 2. Analysis of Intrusion Detection Studies using CIC-2017.

Algorithm \ Dataset		CIC-IDS 2017(CIC-2017)	
ML	Random Forest	O	-
	GNB	-	-
	Decision Tree	-	-
	MLP	-	-
DL	DNN	O	-
	GBT	O	-
	XGBoost	-	O
	CNN	-	-
	RNN	-	-
pre-processing		Convert the dataset into images	Remove socket data
		Data Padding	Remove white space
		P-ZigZag Encoding	Encode label
		-	Normalize data
		-	Replace or Remove missing/infinite data
		-	Remove normal traffic data
evaluation		Binary Classification - DNN	P-ZigZag
		Binary Classification - GBT	
		Multiclass Classification - DNN	OHE
		Multiclass Classification - GBT	
reference		[13]	[14]

Table 3. Analysis of Intrusion Detection Studies using CIC-2018.

Algorithm \ Dataset		CSE-CIC-IDS 2018 (CIC-2018)	
ML	Random Forest	O	-
	GNB	O	-
	Decision Tree	O	-
	MLP	O	-
DL	DNN	-	-
	GBT	-	-
	XGBoost	-	-
	CNN	-	O
	RNN	-	O
pre-processing		Remove normal/noise data	Remove null values and infinite values
		Eliminate unnecessary value after decimal	Convert numerical data into images
		Replace untreatable value	-
evaluation		Classify each Zero-Day attack & benign data	Multiclass Classification - CNN
		Classify mixed Zero-Day attack & benign data	Multiclass Classification - RNN
reference		[15]	Our approach

As ML techniques, they Random Forest, Decision Tree, Gaussian Naïve bayes classifier, Multi-Layer Perceptron (MLP), K-nearest neighbors classifier, and Quadratic discriminant analysis classifier. Table 2 and Table 3 show IDS studies using CIC-2017 and CIC-2018. We can hardly find DL-based IDS studies using CIC-2018. In this work, we suggest an IDS model employing DL techniques.

III. METHODS

3.1. Datasets and features

CSE-CIC-IDS2018(CIC-2018) is a dataset containing network traffic and system logs. CIC-2018 consists of 10 days of sub-datasets collected on different days through injecting 16 types of attacks. This dataset was generated using CICFlowMeter-V3 [24] and contains about 80 types of features. These features provide forward and backward directions of network flow and packets. The size of CIC-2018 is more than 400GB, which is the larger amount than that of CIC-2017. We can develop a DL-based IDS model and evaluate its performance using CIC-2018.

Table 4. Type of injected attacks and amounts of sub-datasets.

Sub-datasets	Type of attacks	Amounts of samples	Total samples
SD - 1	Benign	446,772	1,048,574
	DoS-Hulk	461,912	
	DoS-SlowHTTPTest	139,890	
SD - 2	Benign	663,808	1,044,751
	FTP-BruteForce	193,354	
	SSH-Bruteforce	187,589	
SD - 3	Benign	988,050	1,040,548
	DoS-GoldenEye	41,508	
	DoS-Slowloris	10,990	
SD - 4	Benign	7,313,104	7,889,295
	DDoS-LOIC-HTTP	576,191	
SD - 5	Benign	360,833	1,048,575
	DDOS-HOIC	686,012	
	DDOS-LOIC-UDP	1,730	
SD - 6	Benign	1,042,603	1,042,965
	Brute Force -Web	249	
	Brute Force -XSS	79	
	SQL Injection	34	
SD - 7	Benign	1,042,301	1,042,867
	Brute Force -Web	362	
	Brute Force -XSS	151	
	SQL Injection	53	
SD - 8	Benign	538,666	606,902
	Infilteration	68,236	
SD - 9	Benign	235,778	328,181
	Infilteration	92,403	
SD-10	Benign	758,334	1,044,525
	Bot	286,191	

Table 4 shows a list of the injected attacks and the amount of each sub-dataset we use in this work. We have preprocessed incomplete data such as null values and infinite values.

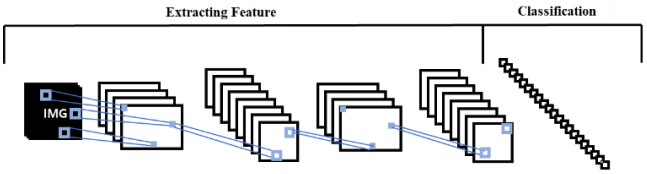
We have extracted 80 types of common features from all the sub-datasets. Using these common features, we can develop and evaluate each attack model in the same environment. We finally choose 79 features except 'Timestamp' as shown in Table 5. The details of the features can be found at the website of CIC-2018 [11].

Table 5. Extracted features for CNN-based intrusion detection.

'Dst Port', 'Protocol', 'Flow Duration', 'Tot Fwd Pkts', 'Tot Bwd Pkts', 'TotLen Fwd Pkts', 'TotLen Bwd Pkts', 'Fwd Pkt Len Max', 'Fwd Pkt Len Min', 'Fwd Pkt Len Mean', 'Fwd Pkt Len Std', 'Bwd Pkt Len Max', 'Bwd Pkt Len Min', 'Bwd Pkt Len Mean',
'Bwd Pkt Len Std', 'Flow Byts/s', 'Flow Pkts/s', 'Flow IAT Mean', 'Flow IAT Std', 'Flow IAT Max', 'Flow IAT Min', 'Fwd IAT Tot', 'Fwd IAT Mean', 'Fwd IAT Std', 'Fwd IAT Max', 'Fwd IAT Min', 'Bwd IAT Tot', 'Bwd IAT Mean', 'Bwd IAT Std', 'Bwd IAT Max', 'Bwd IAT Min', 'Fwd PSH Flags', 'Bwd PSH Flags', 'Fwd URG Flags', 'Bwd URG Flags', 'Fwd Header Len', 'Bwd Header Len', 'Fwd Pkts/s', 'Bwd Pkts/s', 'Pkt Len Min', 'Pkt Len Max', 'Pkt Len Mean', 'Pkt Len Std', 'Pkt Len Var', 'FIN Flag Cnt', 'SYN Flag Cnt', 'RST Flag Cnt', 'PSH Flag Cnt', 'ACK Flag Cnt', 'URG Flag Cnt', 'CWE Flag Count', 'ECE Flag Cnt', 'Down/Up Ratio',
'Pkt Size Avg', 'Fwd Seg Size Avg', 'Bwd Seg Size Avg', 'Fwd Byts/b Avg', 'Fwd Pkts/b Avg', 'Fwd Blk Rate Avg', 'Bwd Byts/b Avg', 'Bwd Pkts/b Avg', 'Bwd Blk Rate Avg', 'Subflow Fwd Pkts', 'Subflow Fwd Byts', 'Subflow Bwd Pkts', 'Subflow Bwd Byts', 'Init Fwd Win Byts', 'Init Bwd Win Byts', 'Fwd Act Data Pkts', 'Fwd Seg Size Min', 'Active Mean', 'Active Std', 'Active Max', 'Active Min', 'Idle Mean', 'Idle Std', 'Idle Max', 'Idle Min', 'Label'

3.2. Design of our CNN model

CNN is the most commonly used deep learning algorithm for image training. In order to develop a CNN-based intrusion model, converting the CIC-2018 dataset into images is required. We convert each labeled data into 13x6 size of images because each data contains 78 features except the 'Label' feature. The 'Label' is used for image classification. A CNN model consists of convolutional layers, max-pooling layers, and a fully connected layer. We can find out the optimal CNN model by organizing those layers along with modeling parameters such as a kernel size, number of kernels, and ratio of dropout. Figure 1 shows our CNN model for CIC-2018.



	Convolution Layer 1	Pooling Layer 1	Convolution Layer 2	Pooling Layer 2	Fully-connected Layer
Num. of kernels	16	-	32	-	-
Size of kernel	3*3	-	3*3	-	-
Size of pooling	-	2*2	-	2*2	-
Dropout	-	0.7	-	0.7	0.5

Fig. 1. Our CNN model and parameters.

We deploy two convolutional layers and the two max-pooling layers behind each convolutional layer. Although the max pooling layer is not mandatory for a CNN model, we deploy the layer because there is very low possibility of losing important features from the max pooling as the converted images only contain numerical data rather than hidden signatures. In addition, we use 'relu' as an activation function for each convolutional layer. In order to reduce overfitting, dropout is applied after each step of the max pooling. Finally, a fully connected layer is deployed behind the last max-pooling layer.

IV. RESULTS AND DISCUSSION

We train each sub-dataset described in Table 4 based on our CNN model. Thirty percent of each sub-dataset is used for the testing set. We set the training parameters as shown in Table 6.

Table 6. Training parameters for our experiments.

Parameters	Value
Optimization algorithm (learning rate)	Adam (0.001)
Size of batch	100
Number of epochs	10

In order to evaluate the performance of our model, we also train the dataset based on RNN model and compare the experimental results with each other. We design the RNN model based on 'vanilla RNN' with 10 units. Figure 2 shows the experimental results of CNN and RNN models. In most sub-datasets, our CNN model has a higher accuracy than that of the RNN model.

The accuracy is measured as follows:

$$Accuracy = \frac{2 \times precision \times recall}{precision + recall}, \quad (1)$$

where $precision = \frac{true\ positives}{true\ positives + false\ positives}$ and $recall = \frac{true\ positives}{false\ negatives + true\ positives}$.

Especially, the accuracies of SD-2, SD-3, SD-5, and SD-9 with CNN are about 10% to 60% higher than that of using RNN.

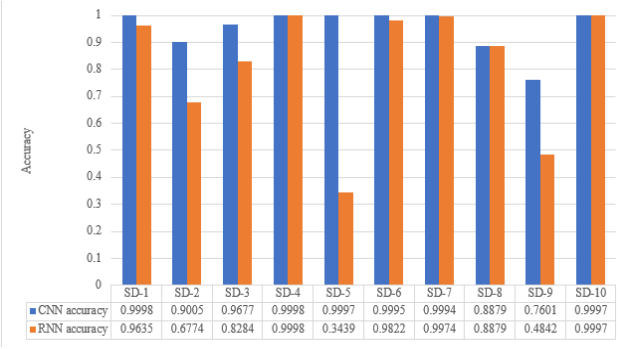


Fig. 2. Comparison of the accuracy of CNN and RNN model.

Although our experimental results show that our CNN model detects attacks in CIC-2018 with high accuracy, we still need to figure out a way of improving the accuracy of each attack. For instance, the accuracy of SD-3 is 0.9677 in Figure 2. According to the confusion matrix of SD-3, however, the accuracies of 'DoS-GoldenEye' and 'DoS-Slowloris' are 0.66 and 0.47 while the accuracy of 'benign' is 0.99 as shown Table 7.

Table 7. Accuracy of each attack with a CNN model.

Sub-datasets	Type of attack	Accuracy
SD-1	Benign	1
	DoS attacks-Hulk	1
	DoS attacks-SlowHTTPTest	1
SD-2	Benign	0.93
	FTP-BruteForce	0.98
	SSH-Bruteforce	0.96
SD-3	Benign	0.99
	DoS attacks-GoldenEye	0.47
	DoS attacks-Slowloris	0.66
SD-4	Benign	1
	DDoS attacks-LOIC-HTTP	1
SD-5	Benign	1
	DDOS attack-HOIC	1
	DDOS attack-LOIC-UDP	1
SD-6	Benign	1
	Brute Force -Web	0.3
	Brute Force -XSS	0.65
	SQL Injection	0.08
SD-7	Benign	1
	Brute Force -Web	0
	Brute Force -XSS	0
	SQL Injection	0
SD-8	Benign	0.94
	Infilteration	0
SD-9	Benign	0.85
	Infilteration	0.35
SD-10	Benign	1
	Bot	1

It means our model has the best performance in classifying benign data because the CIC-2018 dataset provides much more 'benign' data than attack-labeled data.

Here we adjust the ratio of labeled data for the better performance of DL, although the original dataset better represents the real-world network environment and distinguishing anomalous traffic from massive benign traffic in the real network is challenging. In ML and DL, a data preprocessing is an important strategy for high. We preprocessed sub-datasets (SD-3, SD-6, SD-7, SD-8, and SD-9) with low accuracy in attack-labeled data so that the amount of benign data must not be more than five times than that of the smallest amount of attack-labeled data. We then train the datasets using our CNN model. Figure 3 compares the accuracy of each attack before and after the preprocessing considering the data ratio. The experimental results show that the accuracies of most attacks dramatically increase through the preprocessing. We can find out the optimal ratio of benign and attack-labeled data through repeatitive preprocessing and training.

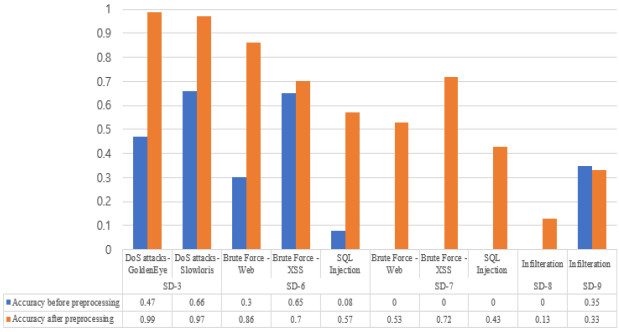


Fig. 3. Comparison of accuracy of before and after preprocessing.

V. CONCLUSION

We have employed DL techniques for intrusion detection. CIC-2018 has been used as an IDS dataset in this work. We have designed a CNN model consisting of two convolutional layers and two max-pooling layers and converted the dataset into images. These images have been trained based on the proposed CNN model and the experimental results showed that our model detects benign and attack data in CIC-2018 with high accuracy. In order to evaluate the performance of our model, we have also trained the dataset using RNN. In the multi-class classification, our CNN model is more accurate than the RNN model when applied to CIC-2018, the latest CIC dataset, using the image-based deep learning method introduced in Tami's work [23]. Furthermore,

we have suggested a way of improving the performance by preprocessing the dataset considering ratio of benign and attack-labeled data. The experimental results showed that the accuracy of attack-labeled data increased through the preprocessing method. In the future, we will train another IDS dataset based on our CNN model and find out the optimal model by reorganizing the convolutional layers along with CNN parameters.

Acknowledgement

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2018R1D1A1B07050543) and also partially supported by special research grant from Seoul Women's University (2019)

References

[1] Jiyeon Kim, Yulim Ahn, and Eunjung Choi, "Network Intrusion Detection using Machine Learning Techniques", in *Proceeding of International Conference on Culture Technology 2019*, August 2019.

[2] Hasan, Md. Al & Nasser, Mohammed & Pal, Biprodip & Ahmad, Shamim, "Support Vector Machine and Random Forest Modeling for Intrusion Detection System (IDS)," *Journal of Intelligent Learning Systems and Applications*, vol. 06, pp. 45-52, 2014.

[3] Mulay, Snehal & Devale, P.R. & Garje, Goraksh, "Intrusion Detection System Using Support Vector Machine and Decision Tree," *International Journal of Computer Applications* vol. 3. 10.5120/758-993, 2010.

[4] Beghad, Rachid, "Training all the KDD data set to classif and detect attacks," *Neural Network World*, vol. 17, pp. 81-91, 2017.

[5] Jia, F. & Kong, L.-Z., "Intrusion Detection Algorithm Based on Convolutional Neural Network," *Beijing Ligong Daxue Xuebao/Transaction of Beijing Institute of Technology*, vol. 37, pp. 1271-1275, 2017.

[6] Yuchen Liu, Shengli Liu and Xing Zhao, "Intrusion Detection Algorithm Based on Convolutional Neural Network", in *Proceeding of the 4th International Conference on Engineering Technology and Application*, 2017.

[7] Jihyun Kim, Howon Kim, An Effective Intrusion Detection Classifier Using Long Short-Term Memory with Gradient Descent Optimization, Proceeding of *the 2017 IEEE International Conference on Platform Technology and Service (PlatCon)*, pp. 1-6, 2017..

[8] R. C. Staudemeyer and C. W. Omlin, "Evaluating performance of long short-term memory recurrent neural networks on intrusion detection data," In *Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference*, pp. 218-224, 2013.

[9] G. Kim, H. Yi, J. Lee, Y. Paek, and S. Yoon, "LSTM-Based System-Call Language Modeling and Robust Ensemble Method for Designing Host-Based Intrusion Detection Systems," *arXiv preprint arXiv:1611.01726*, 2016.

[10] Intrusion Detection Evaluation Dataset (CICIDS2017), https://www.unb.ca/cic/datasets/ids-2017.html

[11] CSE-CIC-IDS2018 on AWS, https://www.unb.ca /cic/datasets/ids-2018.html

[12] Sharafaldin I., Gharib A., Habibi Lashkari A., and Ghorbani A. A.. Towards a reliable intrusion detection benchmark dataset, *Software Networking*, vol. 2017, no. 1, pp. 177–200, 2017.

[13] Faker, Osama & Dogdu, Erdogan, "Intrusion Detection Using Big Data and Deep Learning Techniques," in *Proceedings of the 2019 ACM Southeast Conference*, pp. 86-93. 2019.

[14] Zhang Xueqin, Chen Jiahao, Zhou Yue, Han, Liangxiu, Lin Jiajun, "A Multiple-layer Representation Learning Model for Network-Based Attack Detection," *IEEE Access*. pp. 1-1. 10.1109/ACCESS.2019.2927465, 2019.

[15] Zhou Qianru, Pezaros Dimitrios, "Evaluation of Machine Learning Classifiers for Zero-Day Intrusion Detection -- An Analysis on CIC-AWS-2018 dataset," 2018.

[16] Jackins, V., and D. Shalini Punithavathani. "An anomaly-based network intrusion detection system using ensemble clustering," *International Journal of Enterprise Network Management*, vol. 9.3-4, pp. 251-260, 2018.

[17] Y. X. Meng, "The practice on using machine learning for network anomaly intrusion detection," in *Proceeding of Machine Learning and Cybernetics (ICMLC), 2011 International Conference*, vol. 2, pp. 576-581, IEEE, 2011.

[18] J. Kim, N. Shin, S. Y. Jo, and S. H. Kim, "Method of intrusion detection using deep neural network," in *Proceeding of IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 313-316, 2017.

[19] K. Leung and C. Leckie, "Unsupervised anomaly detection in network intrusion detection using clusters," In *Proceedings of the Twenty-eighth Australasian conference on Computer Science*, vol. 38, pp. 333-342, 2005.

[20] J. J. Davis and A. J. Clark, "Data preprocessing for anomaly based network intrusion detection: A review," *Computers & Security*, vol. 30, no. 6, pp. 353-375, 2011.

[21] Chuanlong Yin, Yuefei Zhu, Jinlong Fei, and Xinzheng He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks", IEEE Access, vol 5, pp. 21954-21961, 2017.

[22] Intrusion detection evaluation dataset (ISCXIDS 2012), https://www.unb.ca/cic/datasets/ids.html

[23] Tamim Mirza, Building an Intrusion Detection System using Deep Learning, https://towardsdata science.com/building-an-intrusion-detection-system-using-deep-learning-b9488332b321, August 2018.

[24] CICFlowMeter, https://www.unb.ca/cic/research/ applications.html#CICFlowMeter

[25] SZEGEDY, Christian, et al. Inception-v4, inception-resnet and the impact of residual connections on learning. In: Thirty-First AAAI Conference on Artificial Intelligence. 2017.

[26] HE, Kaiming et al., "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778, 2016.

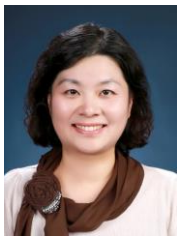
[27] GERS, Felix A.; SCHMIDHUBER, Jürgen; CUMMINS, Fred. Learning to forget: Continual prediction with LSTM. 1999.

[28] VINAYAKUMAR, R., et al., "Deep Learning Approach for Intelligent Intrusion Detection System," *IEEE Access*, vol. 7: 41525-41550, 2019.

Authors

Jiyeon Kim received her BSc and PhD degrees in information security engineering from Seoul Women's University in 2007 and 2013, respectively. She was a postdoctoral research associate in the department of electrical and computer engineering at Carnegie Mellon University from 2014 to 2017. She is a teaching professor at Seoul Women's University since 2019. Her research interests include network security, cloud security, artificial intelligence, cybersecurity and M&S (modeling and simulation) methodology.

Yu-Lim Shin is a undergraduate student in Department of Information Security from Seoul Women's University, Korea. Her research interests include data visualization, malware detection using AI.



Eunjung Choi received her BSc, MS, and PhD degrees in computer engineering from Seoul Women's University in 1997, 2000 and 2005, respectively. She is an associate professor in the department of information security at Seoul Women's University since 2006. Her research interests include artificial intelligence, cybersecurity and big data analysis.