
Proceedings Paper:
Mhamdi, L, McLernon, D orcid.org/0000-0002-5163-1975, El-moussa, F et al. (3 more authors) (2021) A Deep Learning Approach Combining Auto-encoder with One-class SVM for DDoS Attack Detection in SDNs. In: 2020 IEEE Eighth International Conference on Communications and Networking (ComNet). 8th IEEE International Conference on Communications and Networking IEEE ComNet'2020, 27-30 Oct 2020, Hammamet, Tunisia. IEEE . ISBN 978-1-7281-5321-6

https://doi.org/10.1109/ComNet47917.2020.9306073

A Deep Learning Approach Combining Autoencoder with One-class SVM for DDoS Attack Detection in SDNs

Tuan A Tang*, Des McLernon†, Lotfi Mhamdi†, Syed Ali Raza Zaidi†,
Mounir Ghogho‡ and Fadi El-Moussa§
*Da Nang University of Science and Technology, Danang, Vietnam.
†School of Electronic and Electrical Engineering, The University of Leeds, Leeds, UK.
‡International University of Rabat, Morocco.
§BT Security Futures Practice, Adastral Park, Ipswich, IP5 3RE, UK.
Email: tanganhtuan@dut.udn.vn, d.c.mclernon@leeds.ac.uk, l.mhamdi@leeds.ac.uk, s.a.zaidi@leeds.ac.uk,
m.ghogho@leeds.ac.uk and fadiali.el-moussa@bt.com.

***Abstract*—Software Defined Networking (SDN) provides us with the capability of collecting network traffic information and managing networks proactively. Therefore, SDN facilitates the promotion of more robust and secure networks. Recently, several Machine Learning (ML)/Deep Learning (DL) intrusion detection approaches have been proposed to secure SDN networks. Currently, most of the proposed ML/DL intrusion detection approaches are based on supervised learning approach that required labelled and well-balanced datasets for training. However, this is time intensive and require significant human expertise to curate these datasets. These approaches cannot deal well with imbalanced and unlabeled datasets. In this paper, we propose a hybrid unsupervised DL approach using the stack autoencoder and One-class Support Vector Machine (SAE-1SVM) for Distributed Denial of Service (DDoS) attack detection. The experimental results show that the proposed algorithm can achieve an average accuracy of 99.35% with a small set of flow features. The SAE-1SVM shows that it can reduce the processing time significantly while maintaining a high detection rate. In summary, the SAE-1SVM can work well with imbalanced and unlabeled datasets and yield a high detection accuracy.**

***Index Terms*—SDN, software-defined networking, network intrusion detection, autoencoder, one-class SVM, DDoS**

I. Introduction

A. Motivation

In SDN architecture, control and data planes are separated from each other. The logically centralized control plane provides a global network overview that can help to secure the network efficiently. Network attacks (i.e., Port Scan, DDoS, Man-in-the-Middle) can now be detected and mitigated in a real time manner. DDoS attacks have existed for a long time and are becoming more and more complex and directly threaten the network's service availability. These attacks are relatively easy to perform, hard to defend against, and the attacker is rarely traced back because of the distributed nature of DDoS attacks. The attacker launches a DDoS attack using a botnet-group of zombies-to generate a vast amount of traffic against a victim's web server. Zombies or computers that are part of a botnet are usually recruited through the use of worms, Trojan horses or back doors. Defending against DDoS attacks is a challenging issue, and in order to do so, we have to first detect these attacks. There are several methods for detecting DDoS attacks like statistics-based method [1], clustering method [2]. DDoS attacks can be mitigated by some defense mechanisms like a firewall or load balancing. However, these defense mechanisms have their own limitations and efficient. Despite all the effort to tackle these attacks, DDoS attack strategies are constantly evolving, so it is tough to detect and mitigate against sophisticated variants of one of the most common attacks. With the development of new DDoS attacks, the statistic-based method and the clustering method cannot perform well enough, so ML/DL approaches are becoming more and more popular and eficient in detecting these kinds of attacks. Several works [3]–[6] have been carried out to tackle DDoS attacks in SDNs. Most of these works employ supervised learning based approach. This approach requires balanced and labelled datasets for training. However, these datasets are not always available for researchers, and they are especially rare in the context of SDN. Unlike supervised learning, the unsupervised learning approach does not need label information for the data and can address the imbalanced classification problems. One-class Support Vector Machine (OC-SVM) has for a long time been one of the most effective anomaly detection methods and is widely adopted in both research and industrial applications. However, the biggest issues for OC-SVM is the capability to operate with large and high-dimensional datasets due to inefficient features and optimization complexity. As a result, the OC-SVM may not be desirable in big data and high-dimensional anomaly detection applications. Besides, Autoencoder recently emerges as an effective intrusion detection approach in different fields [7]–[9]. In these researches, a reconstruction error is used to detect anomalies. In this paper, we propose an unsupervised hybrid

978-1-7281-5320-9/20/$31.00 © 2020 IEEE

approach combining Stack Autoencoder with OC-SVM (SAE-1SVM) for DDoS attack detection in the SDN.

B. Contribution

Our main contributions are as follows:

- We introduce an unsupervised DDoS attack detection approach in the SDN paradigm using the SAE-1SVM. To the best of our knowledge, this is the first attempt to use the SAE-1SVM for DDoS attack detection in the SDN environment. In our work, the Stack Autoencoder learns the patterns of legitimate traffic and also compresses input data into a lower dimension. This lower-dimensional and higher-level data is now more suitable for the OC-SVM to process.
- Our SAE-1SVM approach yields a detection rate of 99.35% using a minimum number of features compared to other state-of-the-art approaches.
- We also evaluate the computational overhead of the SAE-1SVM. The result shows that our approach has significant potential for real-time intrusion detection.

This paper is organized as follow. In Section II, we introduce some related work. Section III describes our proposed hybrid approach for DDoS attack detection, the CICIDS20127 dataset and evaluation metrics. Section IV shows the performance evaluation of our approach. Finally, conclusions and future work are discussed in section V.

II. Related Work

SDN-based IDSs have been extensively researched recently. One of the earliest approaches for DDoS attack detection in the SDN was proposed in [10]. Braga *et al.* presented a lightweight approach using a Self Organizing Map (SOM) to detect DDoS attacks in the SDN. This approach based on six traffic flow features (Average of Packets per flow, Average of Bytes per flow, Average of Duration per flow, Percentage of Pair-flows, Growth of Single-flows, Growth of Different Ports) gives a quite high detection accuracy. Nam *et al.* [11] proposed an approach combining SOM and K-Nearest Neighbors to detect several kinds of DDoS attacks in SDN. This approach can reduce computational overheads while maintaining a suitable ACC of 98.24%.

Recently, DL has developed as an important research trend in the field of intrusion detection. Tang *et al.* [12] proposed a DL approach for intrusion detection. They achieved a quite promising accuracy of 75.75% using a limited number of flow features. Yin *et al.* [13] proposed a DL approach using recurrent neural networks for detection intrusion. They got an accuracy of 83.28% with their experiments on the NSL-KDD dataset. Fu *et al.* [14] proposed an IDS using Long short term memory RNN (LSTM-RNN). They achieved an accuracy of 97.52% with the NSL-KDD dataset. An autoencoder (AE) - a form of Artificial Neural Network - is extensively exploited by many researchers for anomaly detection in SDNs. Zhang *et al.* [15] proposed a method combining Sparse Autoencoder and Xgboost algorithms to deal with a high-dimensional and unlabelled dataset. They achieved an F1-measure of 91.97%, but their precision is still quite low compared to other state-of-the-art approaches. In [5], the authors proposed a DL based approach using a stacked autoencoder (SAE) for detecting DDoS attacks in the SDN. A non-symmetric deep AE and Random Forest algorithm are combined to detect DDoS attacks in [6]. The authors claim that they can obtain a good classification result whilst significantly reducing the training time.

These proposed methods can detect DDoS attacks with quite high accuracy, but they only support limited types of DDoS attacks and are all supervised learning approaches. From the above approaches, a DDoS detection system which is lightweight and unsupervised is now necessary.

III. Research Methodology/System Description

In this section, we first introduce the SAE-1SVM architecture. Secondly, we describe the CICIDS2017 dataset used in our research. Finally, we explain all the metrics used to evaluate the performance of our proposed approach.

A. SAE-1SVM for DDoS Attack Detection

An AE consists of one input layer, one or more hidden layers and one output layer. The input and output layers always have the same sizes. A general structure for an AE is shown in Fig. 1. The AE has two phases which are encoding and decoding. For encoding process, input data $\mathbf{x}$ is compressed into a low-dimensional representation $\mathbf{h}$ and then the decoder reconstructs the input based on the low-dimensional representation:

$$\mathbf{h} = f(\mathbf{W}\mathbf{x} + \mathbf{b}), \tag{1}$$

$$\mathbf{y} = f(\mathbf{W}'\mathbf{h} + \mathbf{b}'), \tag{2}$$

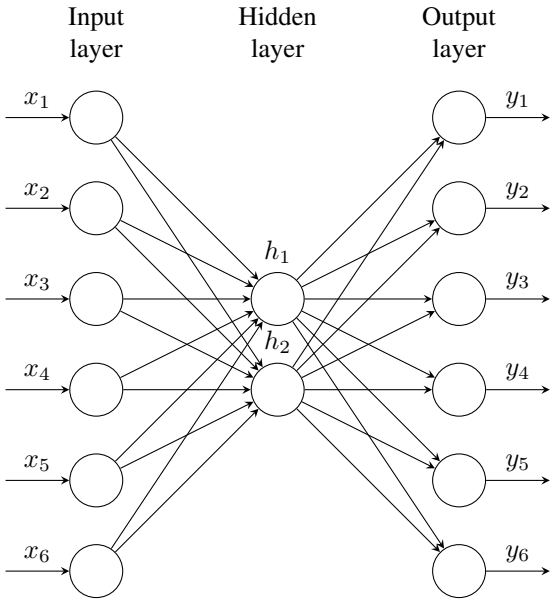

Fig. 1. A General Structure of an AE

where $f(\cdot)$ is a non-linearity activation function, $\mathbf{W}$ and $\mathbf{W}'$ are hidden weight matrices, $\mathbf{b}$ and $\mathbf{b}'$ are biases and $\mathbf{y}$ is the output vector.

The main goal of training the AE is to minimize the difference between the input $\mathbf{x}$ and output $\mathbf{y}$. Therefore, a MSE loss function is used as follows:

$$L(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2^2 . \tag{3}$$

In order to learn feature representations of input data, AEs are stacked successively to form a deep AE (SAE). The learned feature representations will be used as inputs for other classifiers.

The OC-SVM [16] is an unsupervised approach for classification. The OC-SVM tries to learn a hyperplane that best separates all the data points from the origin:

$$f(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) - \rho, \tag{4}$$

where $\phi(\cdot)$ is a feature projection function that maps an input vector $\mathbf{x}$ into a higher dimensional feature space, $\mathbf{w}$ is a decision hyperplane normal vector which is perpendicular to the hyperplane, and ρ is an intercept term. We can obtain $\mathbf{w}$ and ρ by solving an objective function:

$$\begin{aligned} &\min_{\omega,\xi,\rho} \frac{1}{2} \|\mathbf{w}\|^2 + \frac{1}{\nu n} \sum_{i=1}^{n} \xi_i - \rho, \\ &\text{subject to: } \mathbf{w}^T \phi(\mathbf{x}_i) \geq \rho - \xi_i, \xi_i > 0, \end{aligned} \tag{5}$$

where the meta-parameter $\nu \in (0, 1]$ determines the upper bound on the fraction of outliers and the lower bound on the number of training samples used as support vectors, and ξ_i are non-zero slack variables for penalizing the outliers.

By using Lagrangian techniques and a kernel function for the dot-product calculations, the decision function becomes:

$$f(\mathbf{x}) = \sum_{i}^{n} \alpha_i k(\mathbf{x}_i, \mathbf{x}) - \rho, \tag{6}$$

where α_i is a Lagrange multiplier, and $k(\mathbf{x}_i, \mathbf{x}) = \phi(\mathbf{x}_i)^T \phi \mathbf{x}$ is a kernel function. A Radial Basic Function (RBF) kernel is employed in our experiment:

$$k(\mathbf{x}_i, \mathbf{x}) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}\|^2}, \gamma > 0. \tag{7}$$

In this paper, we propose a hybrid approach combining SAE with OC-SVM for DDoS attack detection. Fig. 2 gives a general structure of the proposed SAE-1SVM. The SAE-1SVM is trained with legitimate traffic traces. At first, the legitimate traffic traces are trained with the SAE to extract the low-dimensional representation, and then the low-dimensional representation is trained with OC-SVM for DDoS attack classification. Because the SAE-1SVM is trained with the legitimate traffic only, the anomaly traffic will be considered as outliers which can be easily detected.

B. CICIDS2017 Dataset

As mentioned in [17], most of the current network dataset is out-of-date and not reliable enough, so the CICIDS2017 dataset was proposed as a new benchmark dataset. The CICIDS2017 dataset is claimed to be most up-to-date with all common attacks and real-world traffic. This dataset covers seven types of common attack families (i.e., Brute Force Attack, Heatbleed Attack, Botnet, DoS Attack, DDoS Attack, Web Attack, and Infiltration Attack). This dataset is divided into seven small datasets with different attack scenarios. All of these datasets are labeled and saved in CSV format. Each flow sample in the CICIDS2017 dataset contains 80 flow features which are defined and explained in detail in [18].

In this paper, we choose the Wednesday dataset focusing on DoS, Heartblead, Slowloris, Slowhttptest, Hulk, GoldenEye, and DDoS attacks. These types of attacks are on the rise and are major threats to the SDN architecture. The Wednesday dataset contains 439,683 legitimate traffic and 251,723 anomaly traffic samples.

Since we just focus on the SDN-related flow feature, we extract a subset of 13 SDN-related features out of 80 features of this dataset for our research. Details of these features can be seen in Table. I.

TABLE I
THE CICIDS2017'S FEATURE DESCRIPTION

Feature Name	Description
Source Port	Source port of the flow
Destination Port	Destination port of the flow
Protocol	Protocol type of the flow
Flow Duration	Duration of the flow in microseconds
Total Fwd Packets	Total packets in the forward direction
Total Length of Fwd Packets	Total size of packet in forward direction
Fwd Packet Length Mean	Standard deviation size of packet in forward direction
Flow Bytes/s	Number of flow bytes per second
Flow Packet/s	Number of flow packets per second
Flow IAT Mean	Mean time between two packets sent in the forward direction
Flow IAT Std	Standard deviation time between two packets sent in the forward direction
Fwd Packets/s	Number of forward packets per second
Subflow Fwd Bytes	The average number of packets in a sub flow in the forward direction

Samples in this dataset are normalized into the range of [0-1] by Min-Max scaling. Its mathematical equation is given as:

$$x^{'} = \frac{x - \min(x)}{\max(x) - \min(x)}, \tag{8}$$

where $x^{'}$ is the normalized value, and x is the original value.

C. Evaluation Metrics

The performance and effectiveness of the NIDS are evaluated by several metrics as follows:

- True Positive (TP): the number of anomaly records correctly classified.

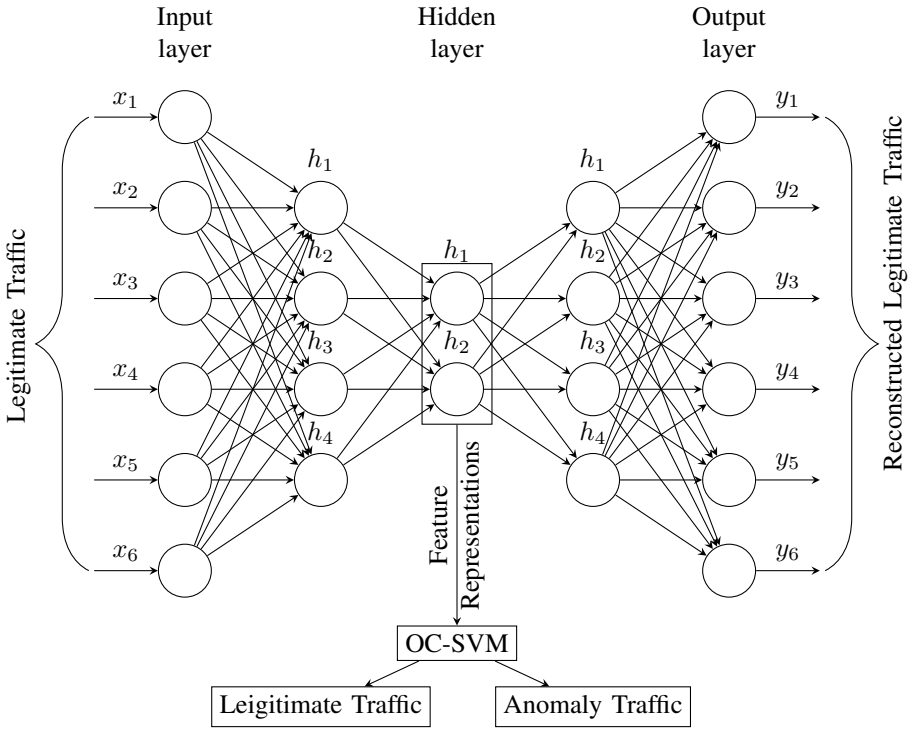

Fig. 2. SAE-1SVM System Detail

- True Negative (TN): the number of normal records correctly classified.
- False Positive (FP): the number of normal records incorrectly classified.
- False Negative (FN): the number of anomaly record incorrectly classified.

For the evaluation purpose, Accuracy (ACC), Precision (P), Recall (R) and F1-measure (F1) metrics are applied. These metrics are calculated as follows:

- Accuracy (ACC): shows the percentage of true detection over total traffic trace:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \times 100\%. \quad (9)$$

- Precision (P): shows how many intrusions predicted by a NIDS are actual intrusions. The higher P then the lower false alarm is:

$$P = \frac{TP}{TP + FP} \times 100\%. \quad (10)$$

- Recall (R): shows the percentage of predicted intrusions versus all intrusions presented. We want a high R value:

$$R = \frac{TP}{TP + FN} \times 100\%. \quad (11)$$

- F1-measure (F1): gives a better measure of the accuracy of a NIDS by considering both the precision (P) and the recall (R). We also aim for a high F value:

$$F1 = \frac{2}{\frac{1}{P} + \frac{1}{R}} \times 100\%. \quad (12)$$

IV. Detection Performance Evaluation

A. Experimental Setup

In our experiment, the SAE architecture is implemented with all hyper-parameters given in Table II. Details about the number of neurons of each network architecture used in this experiment are shown in Table III. For the OC-SVM model, the parameters ν and γ are chosen from the range $\{10^{-10}, 10^{-9},...,10^{0}\}$. After the optimizing process, the parameters in (5) $\nu = 10^{-2}$ and $\gamma = 10^{-2}$ are chosen for the experiment.

TABLE II
The SAE Hyper-parameters

Variable	Parameters
Activation Function	Tanh
Loss Function	Mean Squared Error
Learning Rate	0.001
Batch Size	10
Epoch	1000

TABLE III
Network Architecture Details

Architecture	Input Layer	Hidden Layer	Output Layer
AE	13	2	13
SAE_1	13	6,2,6	13
SAE_2	13	10,8,6,4,2,4,6,8,10	13

B. DDoS Attack Detection with a Hard Threshold

The AE is commonly used to detect anomaly with the idea that behaviors of attacks are different from those of

legitimate traffic. Therefore, the AE will be trained only with the legitimate traffic and then it tries to reconstruct them with the highest ACC. The anomaly traffic is not used for training, so the AE cannot reconstruct them correctly. We can detect anomaly traffic based on this difference. In this section, we analyze the effect of network architecture on the reconstruction performance.

We compare the reconstruction ACC of the AE, SAE_1, and SAE_2 in Table IV. As we can see, the SAE_2 yields the best reconstruction ACC at 98.6%. The AE gives a quite low reconstruction ACC at 85%. With just one hidden layer, we cannot learn good feature representations, so the reconstructed input is just a lossy version of the original inputs. It shows that a deeper SAE can learn feature representations better and then reconstructs the inputs with a higher ACC. Therefore, the SAE_2 will be chosen for further experiments.

TABLE IV
RECONSTRUCTION ACC COMPARISON

Architecture	Reconstruction ACC (%)
AE	85
SAE_1	96
SAE_2	98.6

As in [7], [8], and [9], we also employ the reconstruction error to detect anomalies. The SAE_2 is trained to minimize the reconstruction error, so the error rate should be quite small with the legitimate input traffic. If any anomaly traffic is fed into the SAE_2, the SAE_2 could not recognize it and reconstruct it correctly. In this case, the reconstruction error is higher than normal, and so we can detect the network attack.

In Table V, we compare the performance of different threshold values in terms of ACC, P, R, and F1. As we can see, with a higher threshold, we get a higher detection ACC. However, the other evaluation metrics drop dramatically with high threshold values. The reason for this trend is that more legitimate traffic is classified correctly with a higher threshold, but we also misclassify anomaly traffic. Even with a small threshold of 0.01, the detection P is still worst. If we set a high hard threshold, the false positive rate will increase significantly, and our network will become vulnerable to attacks. If we set a low hard threshold, the false alarm rate will increase and so the NIDS can block the legitimate traffic.

TABLE V
ACCURACY METRICS FOR DIFFERENT THRESHOLDS

Threshold	ACC (%)	P (%)	R (%)	F1 (%)
0.01	54.9	21	85	33.67
0.03	55.3	13.9	4.3	6.5
0.05	58.2	1.7	0.26	0.45

The AE approach with a hard threshold for anomaly detection works quite well in [7], [8], and [9] but it does not perform well in our experiments. This phenomenon can be attributed to the complexity of DDoS attacks in the CICIDS2017 dataset. Some DDoS attacks in this dataset try to mimic behaviours of legitimate traffic, so the reconstruction error rate of both legitimate and anomaly traffic quite close to each other. As a result, they are hard to detect. As seen in this section, the hard threshold approach is not good and efficient enough for DDoS attack detection, so we will consider another approach in the next section.

C. DDoS Attack Detection with the SAE-1SVM

In this section, we analyze the DDoS attack detection performance of the SAE-1SVM. The general architecture of the SAE-1SVM has been described in Fig. 2. In this experiment, we employ the SAE_2 architecture from the previous experiment for feature representation learning. To begin with, we present the detection performance of the SAE-1SVM in term of ACC, P, R, and F1. We compare the performance of SAE-1SVM with classical OC-SVM. We also compare the SAE-1SVM with a DL algorithm combined Convolution Neural Network (CNN) and LSTM proposed in [19]. This work also uses the CICIDS2017 dataset for performance evaluation.

The overall detection performance comparison is depicted in Table VI. According to the experimental results shown in Table VI, we can see that the SAE-1SVM outperforms the OC-SVM in all of the evaluation metrics. Specifically, the SAE-1SVM achieves a much higher P than the OC-SVM. The SAE-1SVM also achieves better results than the CNN+LSTM algorithm. The main reason for this high performance is that the OC-SVM in the SAE-1SVM is trained with the low dimensional representation. The low dimensional representation helps the OC-SVM characterize the network traffic better, so the detection ACC can be improved significantly.

TABLE VI
THE EVALUATION METRIC COMPARISON

Algorithm	ACC (%)	P (%)	R (%)	F1 (%)
OC-SVM	98	96.26	98.21	97.22
CNN+LSTM [19]	98.87	98.89	98.83	98.86
SAE-1SVM	**99.35**	**99.97**	**98.28**	**99.11**

The computational time is an important factor in evaluating the performance of a classifier. Reducing the computational time is also very important. The training and testing times of each algorithm are presented in Table VII. As we can see, the SAE-1SVM consumes significantly less time than OC-SVM in both training and testing processes. The SAE-1SVM is 27 and 6 times faster than the OC-SVM in training and testing respectively. The OC-SVM module in the SAE-1SVM now only processes 2-dimensional inputs compared to 13-dimensional inputs in the original OC-SVM, so the processing time has been reduced significantly. In the SAE-1SVM, the OC-SVM processes more representative but lower-dimensional inputs. Therefore, the SAE-1SVM has excellent potential for real-time NIDS.

TABLE VII
THE TRAINING AND TESTING TIME COMPARISON

Algorithm	Traing Time (s)	Testing Time (s)
OC-SVM	5110	141
SAE-1SVM	189	26

V. CONCLUSION

In this paper, we presented a hybrid unsupervised DL approach for DDoS attack detection. The above results show that our proposed approach has a strong potential in detecting DDoS attacks using limited flow features. The experimental results also show that our SAE-1SVM can deal really well with imbalanced and unlabeled datasets. Although the final results have a quite high false positive rate, the SAE-1SVM can be improved in several ways. Several DL approaches can be applied to the SAE to improve generalizing capability. We can also optimize the OC-SVM by a grid search algorithm. In future research, we will deploy our proposed approach in a real SDN testbed for more detail analysis. Detecting other kinds of network attacks will also be considered in future research.

REFERENCES

[1] A.-S. Kim, H.-J. Kong, S.-C. Hong, S.-H. Chung, and J. W. Hong, "A flow-based method for abnormal network traffic detection," in *Network operations and management symposium, 2004. NOMS 2004. IEEE/IFIP*, vol. 1. IEEE, 2004, pp. 599–612.

[2] P. Casas, J. Mazel, and P. Owezarski, "Unsupervised network intrusion detection systems: Detecting the unknown without knowledge," *Computer Communications*, vol. 35, no. 7, pp. 772–783, 2012.

[3] A. AlEroud and I. Alsmadi, "Identifying cyber-attacks on software defined networks: An inference-based intrusion detection approach," *Journal of Network and Computer Applications*, vol. 80, pp. 152–164, 2017.

[4] S. M. Mousavi and M. St-Hilaire, "Early detection of ddos attacks against sdn controllers," in *Computing, Networking and Communications (ICNC), 2015 International Conference on*. IEEE, 2015, pp. 77–81.

[5] Q. Niyaz, W. Sun, and A. Y. Javaid, "A deep learning based ddos detection system in software-defined networking (sdn)," *arXiv preprint arXiv:1611.07400*, 2016.

[6] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 1, pp. 41–50, 2018.

[7] R. C. Aygun and A. G. Yavuz, "Network anomaly detection with stochastically improved autoencoder based models," in *2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud)*. IEEE, 2017, pp. 193–198.

[8] Y. Kawachi, Y. Koizumi, and N. Harada, "Complementary set variational autoencoder for supervised anomaly detection," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2366–2370.

[9] T. Luo and S. G. Nagarajany, "Distributed anomaly detection using autoencoder neural networks in wsn for iot," in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–6.

[10] R. Braga, E. Mota, and A. Passito, "Lightweight ddos flooding attack detection using nox/openflow," in *Local Computer Networks (LCN), 2010 IEEE 35th Conference on*. IEEE, 2010, pp. 408–415.

[11] T. M. Nam, P. H. Phong, T. D. Khoa, T. T. Huong, P. N. Nam, N. H. Thanh, L. X. Thang, P. A. Tuan, V. D. Loi *et al.*, "Self-organizing map-based approaches in ddos flooding detection using sdn," in *2018 International Conference on Information Networking (ICOIN)*. IEEE, 2018, pp. 249–254.

[12] T. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, "Deep learning approach for network intrusion detection in software defined networking," in *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM) (WINCOM'16)*, Fez, Morocco, Oct. 2016.

[13] C. Yin, Y. Zhu, J. Fei, and X. He, "A deep learning approach for intrusion detection using recurrent neural networks," *IEEE Access*, vol. 5, pp. 21 954–21 961, 2017.

[14] Y. Fu, F. Lou, F. Meng, Z. Tian, H. Zhang, and F. Jiang, "An intelligent network attack detection method based on rnn," in *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*. IEEE, 2018, pp. 483–489.

[15] B. Zhang, Y. Yu, and J. Li, "Network intrusion detection based on stacked sparse autoencoder and binary tree ensemble method," in *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2018, pp. 1–6.

[16] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection," in *Advances in neural information processing systems*, 2000, pp. 582–588.

[17] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proceedings of fourth international conference on information systems security and privacy, ICISSP*, 2018.

[18] "Netflowmeter," http://netflowmeter.ca/netflowmeter.html. Accessed 19 Feb 2019.

[19] A. Pektaş and T. Acarman, "A deep learning method to detect network intrusion through flow-based features," *International Journal of Network Management*, p. e2050.