EasyChair Preprint

№ 638

An Improved Convolutional Neural Network model for Intrusion Detection in Networks

Riaz Ullah Khan, Zhang Xiaosong, Mamoun Alazab and Rajesh Kumar

November 16, 2018

An Improved Convolutional Neural Network model for Intrusion Detection in Networks

Riaz Ullah Khan[1], Xiaosong Zhang[1], Mamoun Alazab[2], and Rajesh Kumar [1]

[1]. Center of Cyber Security, School of Computer Science and Engineering, University of Electronic Science and Technology of China
riazkhan@ieee.org
[2]. College of Engineering, IT and Environment, Charles Darwin University, Australia
mamoun.alazab@cdu.edu.au

Abstract. Network intrusion detection is an important component of network security. Currently, the popular detection technology used the traditional machine learning algorithms to train the intrusion samples, so as to obtain the intrusion detection model. However, these algorithms have the disadvantage of low detection rate. Deep learning is more advanced technology that automatically extracts features from samples. In view of the fact that the accuracy of intrusion detection is not high in traditional machine learning technology, this paper proposes a network intrusion detection model based on convolutional neural network algorithm. The model can automatically extract the effective features of intrusion samples, so that the intrusion samples can be accurately classified. Experimental results on KDD99 datasets show that the proposed model can greatly improve the accuracy of intrusion detection.

Keywords: Network Security, Cyber Security, Intrusion Detection, CNN

1 INTRODUCTION

With the development of Internet technology, more and more physical devices are connected to the Internet. The connection between devices resulted in a large amount of data being generated and saved. The era of "big data" came into being, however, some valuable data is exposed due to lack of protection measures especially when the device transmits data through continuous connection, thus causing huge losses to individuals and even to the whole country [1]. Many machine learning algorithms are used for malware/intrusion detection so far. Khan et. al. [2,3] analysed ResNet and GoogleNet models for malware detection which are based on CNN. Kumar et. al. [4] used CNN model for malicious code detection based on pattern recognition. With the increasing number of networked devices, network systems will become more vulnerable. This gives hackers an opportunity to steal data, user privacy, and trade secrets more easily [5]. Although people have tried their best to protect their important information, due to the complexity of the network system and the richness of attack methods, cyber

attacks continue to occur [6]. Given these circumstances, cyber attack detection methods should be smarter and more efficient than ever before, in order to detect and prevent the growing hacking technology . This paper presents a method of network anomaly detection based on deep learning. Experimental results show that this method can identify daily cyber attacks quickly and efficiently.

This paper studies the network intrusion detection based on convolutional neural networks (CNN) and combines the convolution and pooling operations to better extract the feature relationships between the data. This not only fails to solve the problem of traditional machine learning models. The deep-seated mining of the relationship between data features and the better understanding of the relationships between features than general neural networks.

2 Related work

Intrusion detection technology is an important part of computer network security. The concept of intrusion detection was first proposed by James Nderson in 1980 [7]. The goal of intrusion detection is to correctly identify abnormal network behavior. The current popular intrusion detection method is to reduce the error rate by using different machine learning techniques. Chung et al. [8], constructed a set of human intrusion detection models by combining various machine learning algorithms, such as support vector machines, Bayesian classification, and decision trees. Pan et al. [9], proposed a hybrid machine learning technique combining Zhi-Mean and SVM to detect attacks. Shin et al. [10], used the bogey-means algorithm to calculate the similarity between data, and by adjusting the parameters. Azab et. al. [11] proposed machine learning techniques for *Zeus V1.x*, *Zeus V2.x* and *benign HTTP* traffic detection in networks. Bamakan et al. [12], and Mohamad Tahir et al. [13], applied neural network to detect intrusion in the network. Zhao [14], proposed the LSSVM model for network intrusion detection. Jha et al. [15], used hidden Markov models to study network intrusion detection. Bamakan et al. [12], used KSVC to classify network intrusions. Horng et al. [16], applied the SVM method to IDS. Traditional machine learning methods are very effective in intrusion detection, but they also have limitations, because the traditional machine learning technology needs to artificially construct sample features. Its performance is dependent on its quality. In order to solve this problem, researchers have introduced deep learning techniques. Gao et al. [17], applied deep trust network in intrusion detection and achieved better results than other traditional machine learning methods. Raman [18], applied probabilistic neural networks to detection techniques. Peddabachigari [19], proposed a hybrid intrusion detection model based on deep learning and verified that the model is more efficient than traditional machine learning methods.

3 Model design

3.1 Architecture of Convolutional Neural Network

As shown in Figure 1, the convolutional neural network structure is composed of input, convolutional layer, pooled layer, fully connected layer and output layer. The convolutional neural networks with different structures have different numbers of convolution and pooling layers. Assuming that the input feature of the convolutional neural network is X, and the feature map of the i-th layer is $M_i(M_\bigcirc = X)$, the convolution process can be expressed as:

$$M_i = f(M_{i-1} \otimes w_i + b_i) \tag{1}$$

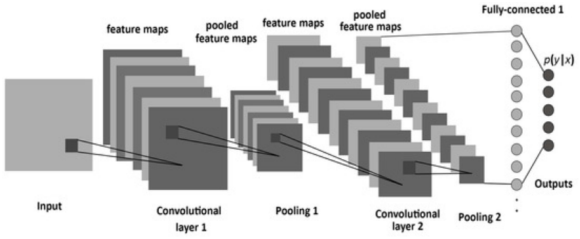

Fig. 1. Convolutional neural network classical structure diagram

W_i is the weight vector of the convolution kernel of the i-th layer, the operation symbol $\bigotimes$ represents the convolution operation, and is the offset vector of the i-th layer, and (z) is the excitation function. In the convolution process, the convolution kernel constructs new features by repeating the convolution operation with the input features. When convolving with a convolutional kernel, the principle of "parameter sharing" is followed. That is, sharing the same weights and offsets makes the number of parameters of the entire neural network greatly reduced.

The pooling layer usually samples the feature map according to different sampling rules after the convolution layer. Assume M For the input of the pooling layer, H_i is the output of the pooling layer, then the pooling layer can be represented as:

$$H_1 = subsampling(M_{i-1}) \tag{2}$$

3.2 Intrusion Detection Model Framework

Figure 2 is a network intrusion detection framework based on convolutional neural network algorithm used in this paper. It can be seen in Figure 2, that the framework mainly consists of three steps:

Step 1. Data preprocessing: It mainly converts symbolic data into numerical data, and then normalizes the data. Details are given section 4.2.

Step 2. Training and Feature Extraction: Use our designed CNN model for data training and feature extraction.

Step 3. Classification: Use the softmax classifier to classify and get the classification results.

We designed an efficient convolutional neural network. The entire network consists of three hidden layers. Each hidden layer contains a convolutional layer and a pooling layer. The number of convolution kernels is different for each hidden layer. The network uses 2×2 convolution kernels and 2×2 pooled cores to enhance performance by continuously deepening the network structure. The number of convolution kernels in each convolutional layer in the network is different. The more the number of post-convolution kernels (32-64-128), this practice of increasing the number of convolution kernels maps the original features into high-dimensional space, thereby enhancing the ability to learn features.

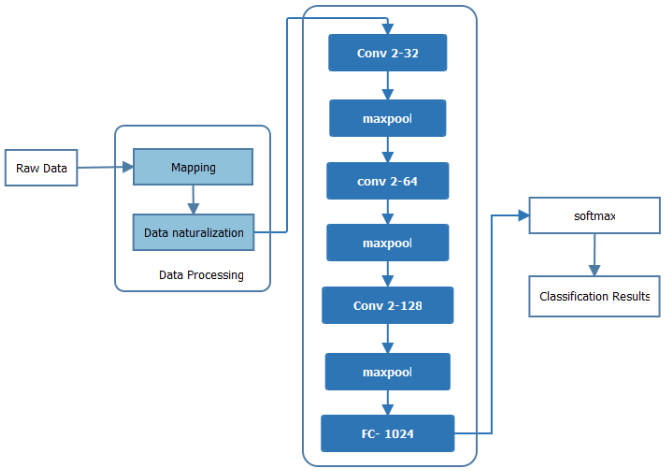

Fig. 2. Proposed Model for Intrusion Detection

4 Experiments and analysis

4.1 Dataset

The data set used in this paper is the KDD99 data set. The data set divides network intrusion into 5 categories: Normal, DOS, R2L, U2R, and probing.Each behavior is represented by its features. This paper uses KDD99 dataset to train the model. This data set contains 494021 training samples and 311029 test samples, The distribution of various types of invasion is shown in Table 1.

Table 1. Distribution of KDD99 datasets

Attack Type	Training Sets	Testing Sets
Normal	97278	60593
DOS	391 458	229 853
R2L	1126	16189
U2R	32	228
probe	4107	4166

4.2 Data preprocessing

The KDD99 dataset contains 41 features per record. It contains 38 numerical features and 3 symbolic features. For these features, the dataset needed to be processed separately.

1) Numerical characterization of symbolic features: For the three symbolic features we used the one-hot method to digitize, for example, for the protocol-type feature, it contains 3 characters No. i.e., tcp, udp, icmp, we convert it to [1, o, o], [o, l, o], [o, o, 1], so we converted the 1D vector into a 3D vector. Similarly, for the service feature, it contains 70 symbols and we converted it to a 70-dimensional vector. For flag features, it packs with 11 symbols, we converted to 11-dimensional vectors. Through the above operation, we map the three symbolic features into 84-dimensional vectors.

2) Normalization of numerical features: For numerical features, due to the different dimensions, the magnitudes of the numerical features are very different. Therefore, in order to eliminate the influence of dimension differences, it is necessary to carry out numerical values. The normalized formula is as follows:

$$x = \frac{x - Min}{Max - Min} \tag{3}$$

4.3 Experimental evaluation standard

This experiment uses accuracy (AC) as an evaluation index to measure the effect of the model. AC formula is as follows:

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

Among them, TP is the number of samples of attack behaviors that are correctly classified;

TN is the number of samples of normal behaviors that are correctly classified;

FP is the number of samples of normal behaviors that are misclassified;

FN is the number of samples of misclassified attack behaviors.

4.4 Analysis of Experimental Results

The data used in this article is a 10% KDD99 dataset use Accuracy (AC) as a verification indicator. The convolution kernel has a convolution kernel whose

Table 2. Metrics for Prediction

		Predicted	
		Normal	Attack
Actual	Normal	TN	FP
	Attack	FN	TP

length and width are both set to 2, the step length is set to 1, the length and width of the pooling layer are both set to 2, the step length is set to 2, and the pooled layer adopts max. The pooling algorithm performs down sampling using the Adam optimization algorithm to optimize the loss function. We observe changes in accuracy by setting the number of epochs in the convolutional neural network. From figure 3 it can be seen that with the continuous increase in the number of epoch, accuracy is rising. It is also observed in Figure 3 that the accuracy is rising when we increase the number of epochs.

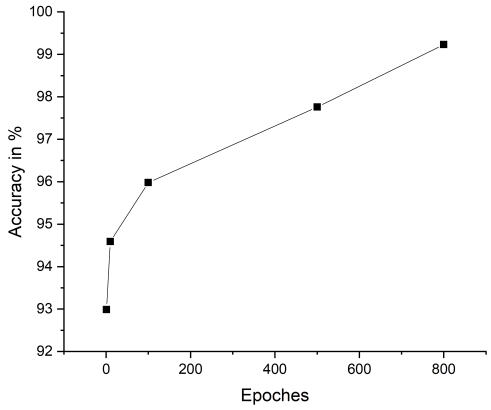

Fig. 3. Increasing accuracy of test results by increasing the number of epochs

Table 3 compares the detection effect by accuracy of the model and SVM, DBN and CNN algorithms. It can be seen that the detection effect of the improved CNN model is higher than that of other algorithms. Therefore, the proposed model is effective.

5 Conclusion

The application of convolutional neural network algorithm in intrusion detection is a new idea. This paper proposes a method combining convolutional neural network algorithm and softmax algorithm. The experimental results show that this

Table 3. Comparison of SVM, DBN and Improved CNN models

Model	Accuracy %
SVM	98.20
DBN	98.59
Improved CNN	99.23

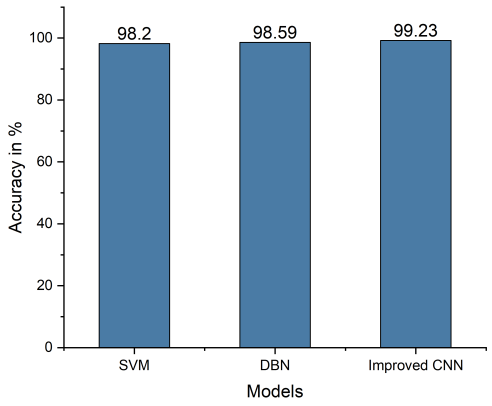

Fig. 4. Comparison of the Improved CNN and other models

model can improve the accuracy of human intrusion detection and improve the performance of human invading detection system. It is observed in the results that the accuracy is increasing when we increase the number of epochs. It is also observed that the proposed model performed better as compare to SVM and DBN models.

Acknowledgment: This research was supported by the National Natural Science Foundation of China under grant No. 61572115.

References

1. S. Venticinque et A. Amato, Smart sensor and big data security and resilience, *in Security and Resilience in Intelligent Data-Centric Systems and Communication Networks*, p. 123–141, Elsevier, 2018.
2. R. U. Khan, X. Zhang et R. Kumar, Analysis of resnet and googlenet models for malware detection, *Journal of Computer Virology and Hacking Techniques*, Aug 2018.
3. R. U. Khan, X. Zhang, R. Kumar et E. O. Aboagye, Evaluating the performance of resnet model based on image recognition, *in Proceedings of the 2018 International Conference on Computing and Artificial Intelligence*, ICCAI 2018, (New York, NY, USA), p. 86–90, ACM, 2018.

4. R. Kumar, Z. Xiaosong, R. U. Khan, I. Ahad et J. Kumar, Malicious code detection based on image processing using deep learning, *in Proceedings of the 2018 International Conference on Computing and Artificial Intelligence*, ICCAI 2018, (New York, NY, USA), p. 81–85, ACM, 2018.
5. J. West, A prediction model framework for cyber-attacks to precision agriculture technologies, *Journal of Agricultural & Food Information*, p. 1–24, 2018.
6. W. Leonard, Resilient cyber-secure systems and system of systems: Implications for the department of defense, *in Disciplinary Convergence in Systems Engineering Research*, p. 145–156, Springer, 2018.
7. J. NDERSON, Computer security threat monitoring and surveillance, *Rapport Technique, James P. Anderson Company, Fort Washington, Pennsylvania*, 1980.
8. S. Chung et K. Kim, A heuristic approach to enhance the performance of intrusion detection system using machine learning algorithms, *in Proceedings of the Korea Institutes of Information Security and Cryptology Conference (CISC-Wâ15)*, 2015.
9. X. Pan, Y. Luo et Y. Xu, K-nearest neighbor based structural twin support vector machine, *Knowledge-Based Systems*, vol. 88, p. 34–44, 2015.
10. D. H. Shin, K. K. An, S. C. Choi et H.-K. Choi, Malicious traffic detection using k-means, *The Journal of Korean Institute of Communications and Information Sciences*, vol. 41, no. 2, p. 277–284, 2016.
11. A. Azab, M. Alazab et M. Aiash, Machine Learning Based Botnet Identification Traffic, *in 2016 IEEE Trustcom/BigDataSE/ISPA*, p. 1788–1794, IEEE, aug 2016.
12. S. M. H. Bamakan, H. Wang et Y. Shi, Ramp loss k-support vector classification-regression; a robust and sparse multi-class approach to the intrusion detection problem, *Knowledge-Based Systems*, vol. 126, p. 113–126, 2017.
13. H. Mohamad Tahir, W. Hasan, A. Md Said, N. H. Zakaria, N. Katuk, N. F. Kabir, M. H. Omar, O. Ghazali et N. I. Yahya, Hybrid machine learning technique for intrusion detection system, 2015.
14. Z. Fuqun, Detection method of lssvm network intrusion based on hybrid kernel function, *Modern Electronics Technique*, vol. 21, p. 027, 2015.
15. S. Jha, K. M. Tan et R. A. Maxion, Markov chains, classifiers, and intrusion detection., *in csfw*, vol. 1, p. 206, Citeseer, 2001.
16. S.-J. Horng, M.-Y. Su, Y.-H. Chen, T.-W. Kao, R.-J. Chen, J.-L. Lai et C. D. Perkasa, A novel intrusion detection system based on hierarchical clustering and support vector machines, *Expert systems with Applications*, vol. 38, no. 1, p. 306–313, 2011.
17. N. Gao, L. Gao, Q. Gao et H. Wang, An intrusion detection model based on deep belief networks, *in Advanced Cloud and Big Data (CBD), 2014 Second International Conference on*, p. 247–252, IEEE, 2014.
18. M. G. Raman, N. Somu, K. Kirthivasan et V. S. Sriram, A hypergraph and arithmetic residue-based probabilistic neural network for classification in intrusion detection systems, *Neural Networks*, vol. 92, p. 89–97, 2017.
19. S. Peddabachigari, A. Abraham, C. Grosan et J. Thomas, Modeling intrusion detection system using hybrid intelligent systems, *Journal of network and computer applications*, vol. 30, no. 1, p. 114–132, 2007.