Detection and Recognition of Abnormal Data Caused by Network Intrusion Using Deep Learning

Yan Jian
Henan Polytechnic, Zhengzhou, Henan 450046, China
E-mail: rhj995@yeah.net

Xiaoyang Dong
Henan Logistics Vocational College, Zhengzhou, Henan 453514, China

Liang Jian
Zhengzhou Vocational University of Information and Technology, Zhengzhou, Henan 450046, China

Keywords: deep learning, network intrusion, abnormal data, detection and recognition

Received: July 9, 2021

Based on deep learning, this study combines sparse autoencoder (SAE) with extreme learning machine (ELM) to design an SAE-ELM method to reduce the dimension of data features and realize the classification of different types of data. Experiments were carried out on NSL-KDD and UNSW-NB2015 data sets. The results show that, compared to the K-means algorithm and the SVM algorithm, the proposed method has higher performance. On the NSL-KDD data set, the average accuracy rate of the SAE-ELM method was 98.93%, the false alarm rate was 0.17%, and the missing report rate was 5.36%. , The accuracy rate of the SAE-ELM method on the UNSW-NB2015 data set was 98.88%, the false alarm rate was 0.12%, and the missing report rate was 4.31%. The results show that the SAE-ELM method is effective in the detection and recognition of abnormal data and can be popularized and applied.

Povzetek: S pomočjo metod globokega učenja je metoda sposobna prepoznati nenormalne podatke v mreži kot posledico vdora.

1 Introduction

With the expansion of the network and the increasing volume of data [1], the traditional methods are increasingly unable to meet the needs of detection and identification of abnormal data, and cannot achieve effective defense of the network. The detection and recognition of abnormal data can be regarded as a classification problem. Methods such as machine learning were widely used in recognition of abnormal data [2] and achieved good results. Mitchell et al. [3] detected the medical network physical system with a behavior-based method. Through experiments, they found that the method could deal with more covert attacks with a high detection rate. Hosseini et al. [4] designed a method based on multi-criteria linear programming and particle swarm optimization. They performed experiments on the KDD CUP 99 and found that it had obvious advantages in accuracy and computing time. Wei et al. [5] used different neural networks to obtain the characteristics of the data for detection and carried out experiments on DARPA 1998 and ISCX2012. The results showed that the method had a good detection rate. Dubey et al. [6] designed a hybrid method based on K-means, naive Bayes, and back-propagation (BP) neural network. They carried out experiments on KDD CUP99 to verify the performance of the method. At present, in the face of massive data, the performance of detection and recognition is not good enough and is greatly affected by the size of the data. Intelligent methods such as deep learning have good detection ability for multi-dimensional dynamic network data; therefore, this paper used deep learning to detect and recognize abnormal data and verified the reliability of the method. This work makes some contributions to further improving abnormal data detection and recognition ability and realizing network security.

2 Detection and recognition method based on deep learning

2.1 Feature extraction based on sparse autoencoder

Autoencoder (AE) [7] is a deep learning network structure. It is assumed that the input of the encoder is I, the middle layer is Z, and the output is O. The purpose of AE is to make I≈O. In this process, the output of the encoder can be written as:

$$Z = f(I) = f_I(W + b_I)$$

The output of the decoder can be written as:

$$O = g(Z) = g_Z(W^T + b_Z)$$

where f_I and g_Z are activation functions, W is an initial weight, b_I is a forward bias, and b_Z is a reverse bias. AE minimizes reconstruction error by training {W,b_I,b_Z }:

$$E = \sum_{x \in I} J\left(x, g\left(f(x)\right)\right)$$

where J refers to the reconstruction error function. This study uses the mean square error loss function:

$$L(x) = \|x - y\|^2$$

Sparse autoencoder (SAE) [8] is obtained by adding a sparsity limitation to AE, which enables it to give deeper features, i.e., let the node's output be as close to 0 as possible. It is assumed that the mean value of the activation degree of node j in the middle layer is:

$$\hat{\rho}_j = \frac{1}{m}\sum_{i=1}^{m}\left[a_j^{(2)}\left(x^{(i)}\right)\right]$$

where m is the number of data and a_j^((2)) is the output activation value of node j, whose input is x. In the sparsity limitation, to make ρ^_j as close as possible to 0, a decimal ρ that approaches 0 is introduced as the sparsity parameter, and Kullback-Leible divergence is used to perform regularized constraint on the network. The global loss function of the network is written as:

$$J_{sparse}(W,b) = J(W,b) + \beta\sum_{j=1}^{s_2} KL\left(\rho\middle\|\hat{\rho}_j\right)$$

$$KL\left(\rho\middle\|\hat{\rho}_j\right) = \rho log\frac{\rho}{\rho_j} + (1-\rho)log\frac{1-\rho}{1-\rho_j},$$

where s_2 refers to the number of neurons in the middle layer.

2.2 Detection and recognition based on extreme learning machine

In the learning process, an extreme learning machine (ELM) [9] can achieve the desired effect by calculating the output weight only, showing a high learning speed [10]. For a given training sample {x_i,y_i }_(i=1)^N, it is assumed that the number of nodes in the hidden layer is L, then

$$o_j = \sum_{i=1}^{L}\beta_i g\left(W_i \cdot X_j + b_i\right)$$

where g(x) is an activation function, W_i is an input weight, β_i is an output weight, and b_i is a bias.

The objective of the network is to minimize the output error:

$$\sum_{j=1}^{N}\|o_j - y_j\| = 0$$

It can be expressed as Hβ=T by a matrix, where H refers to the node's output in the hidden layer, T is the expected output, and β is the output weight. The solution is:

$$\beta = H^{+}T$$

where H^+ is the Moore-Penrose generalized inverse of H [11].

In the SAE-ELM method designed in this paper, firstly, the dimension of features is reduced by the SAE method. In a given sample set, {(X^1, 〖Y 〗^1),(X^2,Y^2),⋯,(X^i,Y^i)}, X^i is the feature vector, and X^i is the labeled vector. After the dimensionality reduction, a new {X_i,Y_i } is obtained. Then, it was detected by the ELM method.

3 Experimental analysis

3.1 Experimental setup

The experimental platform was MATLAB2014a. The operating system was Win10 64 bits. The processor was Intel(R)Core(TM)i7-9700K CPU @3.6Hz with16GB memory. Nvida RTX 2060 (6 GB) graphic card was used. The activation function was sigmod and the sparsity parameter was set to 0.25. There were 14 hidden layers used.

The experimental data sets were NSL-KDD and UNSW-NB2015. NSL-KDD is a benchmark data set [12, 13], which is usually used to estimate the behavior of the network data. Each data has 41 features; there are one class of normal data and four classes of abnormal data(DOS, Probe, R2L, and U2R). Experiments were carried out with 125973 data in KDDTrain, as shown in Table 1.

	Training set	Test set
Normal	53875	13468
DOS	36742	9185
Probe	9352	2331
R2L	797	198
U2R	42	10
Total	100781	25192

Table 1: NSL-KDD data set.

UNSW-NB2015 is a relatively new data set [14], recording the normal activities and attack behaviors of real modern networks [15], which are as follows:

(1) normal: normal data;

(2) fuzzers: pause the network by providing randomly generated data;

(3) analysis: attacks including port scanning and spam;

(4) backdoors: access the computer by bypassing the system security mechanism;

(5) DoS: users cannot use the server or network resources;

(6) exploits: attack the host through vulnerabilities;

(7) generic: an attack used for password countermeasure;

(8) reconnaissance: collect the information of the victim's host and attack it;

(9) shellcode: attack the computer through vulnerabilities of software;

(10) worms: attackers copy themselves and propagate to other computers.

219160 data in one subset used in the experiment, as shown in Table 2.

	Training set	Test set
Normal	35983	53122
Fuzzers	4885	16852
Analysis	69	636
Backdoors	83	443
DoS	1452	3399
Exploits	8281	21595
Generic	18830	39754
Reconnaissance	3217	8874
Shellcode	378	1133
Worms	44	130
Total	73222	145938

Table 2: UNSW-NB2015 data set.

3.2 Evaluation index

(1) Accuracy: A_C=(T_P+T_N)/(T_P+T_N+F_P+F_N),
(2) false alarm rate: F_A=F_P/(T_N+F_P),
(3) missing report rate: M_A=F_N/(T_P+F_N),
where T_P refers to the number of abnormal data that are classified as abnormal, T_N refers to the number of normal data that are classified as normal, F_P refers to the number of normal data that are classified as abnormal, and F_N refers to the number of abnormal data that are classified as normal.

3.3 Experimental results

Firstly, the binary classification experiment was carried out on NSL-KDD, and the results were compared with the support vector machine (SVM) algorithm [16] and the K-means algorithm [17], as shown in Figure 1.

It was seen from Figure 1 that the SAE-ELM method had the best performance in detecting and recognizing abnormal data. The accuracy A_c of the K-means, SVM, and SAE-ELM algorithms was 74.64%, 86.48%, and 95.64%, respectively; the A_c of the SAE-ELM algorithm was 21.02% higher than the K-means algorithm and 9.16% higher than the SVM algorithm. The F_A of K-means, SVM, and SAE-ELM algorithms was 4.67%, 1.89%, and 0.45%, respectively; the F_A of the SAE-ELM algorithm was 4.22% lower than that of the K-means algorithm and 1.44 % lower than that of the SVM algorithm. The M_A of the SAE-ELM algorithm was 7.41 % lower than that of the K-means algorithm and 4.84 % lower than that of the SVM algorithm. The above results verified that the SAE-ELM algorithm was reliable.

	Accuracy/%	False alarm rate/%	Missing report rate/%
Normal	99.67	0.18	7.42
DOS	99.34	0.27	6.43
Probe	98.77	0.24	5.68
R2L	98.56	0.12	4.21
U2R	98.33	0.02	3.08
Average	98.93	0.17	5.36

Table 3: Results of the five-classification experiment on the NSL-KDD data set.

Then, a five-classification experiment was carried out on the NSL-KDD data set, as shown in Table 3.

It is clear from Table 3 that the SAE-ELM algorithm had the best performance in detecting and recognizing normal data but performes poorly in detecting and recognizing U2R. The samples of U2R were the least among the different kinds of data, which led to the insufficient training of the algorithm. The amount of normal data was the largest; thus, the accuracy of the detection and recognition of normal data was the highest (99.67%). The average A_c, F_A, and M_A of the SEA-ELM algorithm was 98.93%, 0.17%, and 5.36 %, respectively.

A binary classification experiment was carried out on UNSW-NB2015 and compared to SVM and K-means algorithms, as shown in Figure 2.

It can be observed from the Figure 2 that the performance of the SAE-ELM method was the best on the NSW-NB2015 data set. The A_c of the three methods was 80.27%, 92.36%, and 99.42%, respectively. The A_c of the SAE-ELM method was 19.15% higher than the SAE-ELM method and 7.06% higher than that of the SVM method. The F_A of the SAE-ELM algorithm was 2.85% lower than that of the K-means algorithm and 0.95% lower than the SVM algorithm. The M_A of the SAE-ELM method was 6.65% lower than that of the K-means

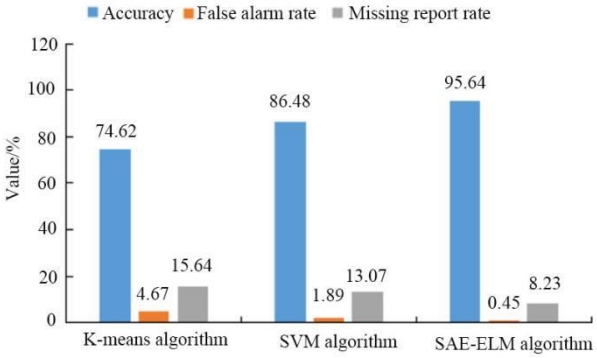

Figure 1: Comparison of results of the binary classification experiment on the NSL-KDD data set.

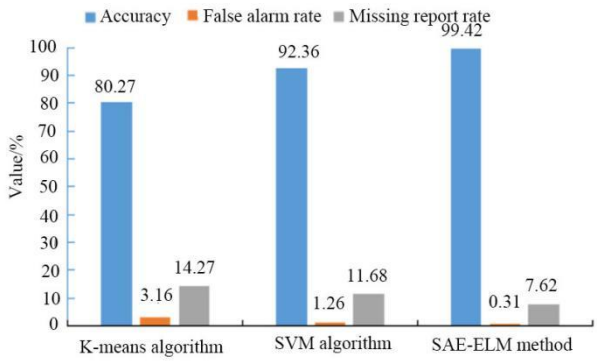

Figure 2: Comparison of results of the binary experiment on the UNSW-NB2015 data set.

	Accuracy/ %	False alarm rate/%	Missing report rate/%
Normal	99.62	0.21	6.48
Fuzzers	98.89	0.16	4.87
Analysis	98.52	0.12	3.55
Backdoors	98.48	0.08	3.51
DoS	98.64	0.11	4.11
Exploits	99.31	0.18	5.12
Generic	99.46	0.17	5.36
Reconnaissance	98.76	0.11	4.36
Shellcode	98.61	0.07	3.61
Worms	98.47	0.01	2.12
Average	98.88	0.12	4.31

Table 4: Results of the multi-classification experiment on the UNSW-NB2015 data set.

algorithm and 4.06% lower than that of the SVM algorithm.

Finally, the polyphenols experiment was carried out on the NSW-NB2015 data set using the SAE-ELM algorithm, as shown in Table 4.

It can be observed from the Table 4 that, similar to the NSL-KDD data set, the SAE-ELM method had better detection and recognition performance in the category with more samples. For the attack type with less number, A_c was relatively small, but all above 95%. The average A_c of the SAE-ELM algorithm was 98.88%, the average F_A was 0.12 %, and the average M_A was 4.31% on the UNSW-NB2015 data set, showing that the SAE-ELM algorithm had a good performance.

4 Discussion

With the development of society, network security has been paid more and more attention [18]. As the data in the network is becoming more and more massive, high-dimensional, and changeable, the traditional detection and protection methods have not been able to meet the current network security needs [19]. Therefore, it is of great significance to find effective detection and identification methods for abnormal data [20]. Deep learning methods have been widely used in image recognition [21], speech recognition [22], intelligent translation [23], etc., which can achieve high classification accuracy in large databases. Therefore, this paper analyzed the application of deep learning in the detection and recognition of abnormal data to know whether it can detect and recognize abnormal data quickly and accurately.

It was found from the experiments on NSL-KDD and UNSW-NB2015 data sets that the A_c and F_A of the SAE-ELM method were better than K-means and SVM algorithms. For the detection and recognition of abnormal data, only larger A_c, small F_A , and low M_A can meet the actual needs. First, in the binary classification experiment, the A_c of the SAE-ELM method was above 98% on the two data sets, and the F_A and M_A were small. In the multi-classification experiment, the average A_c, F_A, and M_A of the SAE-ELM method were 98.93%, 0.17%, and 5.36%, respectively. On the UNSW-NB2015 data set, the A_c, F_A, and M_A of the SAE-ELM method were 98.88%, 0.12%, and 4.31%, respectively. The two experiments showed good performance of the SAE-ELM method.

Although some results were attained on the recognition and detection of abnormal data, more research is still needed:

(1) the usability of additional deep learning methods should be studied;

(2) the actual network operational data should be collected for the detection and identification.

5 Conclusion

Based on deep learning, this paper analyzes the detection and recognition of abnormal data, introducesan SAE-ELM method, and presents carried out experiments on NSL-KDD and UNSW-NB2015 data sets. It was found that the SAE-ELM method has high accuracy and good performance in detecting and recognizing of the abnormal data, which can be further promoted and applied in practice.

References

[1] Omar Y. Al-Jarrah, Omar Alhussein, Paul D. Yoo, Sami Muhaidat, Kamal Taha, and Kwangjo Kim. Data Randomization and Cluster-Based Partitioning for Botnet Intrusion Detection. IEEE Transactions on Cybernetics, 46(8):1796-1806, 2015. https://doi.org/10.1109/TCYB.2015.2490802.

[2] Anna L. Buczak, and Erhan Guven. A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection. IEEE Communications Surveys & Tutorials, 18(2):1153-1176, 2017. https://doi.org/10.1109/COMST.2015.2494502.

[3] Robert Mitchell, and Ing-Ray Chen. Behavior Rule Specification-Based Intrusion Detection for Safety Critical Medical Cyber Physical Systems. IEEE Transactions on Dependable & Secure Computing, 12(1):16-30, 2015. https://doi.org/10.1109/TDSC.2014.2312327.

[4] Seyed Mojtaba Hosseini Bamakan, Behnam Amiri, Mahboubeh Mirzabagheri, and Yong Shi. A New Intrusion Detection Approach Using PSO based Multiple Criteria Linear Programming. Procedia Computer Science, 55:231-237, 2015. https://doi.org/10.1016/j.procs.2015.07.040.

[5] Wei Wang, Yiqiang Sheng, Jinlin Wang, Xuewen Zeng, Xiaozhou Ye, Yongzhong Huang, and Ming Zhu. HAST-IDS: Learning Hierarchical Spatial-Temporal Features Using Deep Neural Networks to Improve Intrusion Detection. IEEE Access, 6(99):1792-1806, 2018. https://doi.org/10.1109/ACCESS.2017.2780250.

[6] Shreya Dubey, and Jigyasu Dubey. KBB: A hybrid method for intrusion detection. International Conference on Computer, NY, India, 1-6, 2015. https://doi.org/10.1109/IC4.2015.7375704.

[7] Samed Sivaslioglu, Ferhat Ozgur Catak, and Kevser ahinba. A generative model based adversarial security of deep learning and linear classifier models. Informatica 45(1):33-64, 2021. https://doi.org/10.31449/inf.v45i1.3234.

[8] Xiong Luo, Yang Xu, Weiping Wang, Manman Yuan, Xiaojuan Ban, Yueqin Zhu, and Wenbing Zhao. Towards Enhancing Stacked Extreme Learning Machine With Sparse Autoencoder by Correntropy. Journal of the Franklin Institute, 355(4): 1945-1966, 2017. https://doi.org/10.1016/j.jfranklin.2017.08.014.

[9] Subhadra Mishra, Debahuti Mishra, Pradeep Kumar Mallick, Hari Gour, and Sachin Kumar. A Novel Borda Count Based Feature Ranking and Feature Fusion Strategy to Attain Effective Climatic Features for Rice Yield Prediction. Informatica, 45(1):13-31, 2021. https://doi.org/10.31449/inf.v45i1.3258.

[10] Yimin Yang, and Q. M. Jonathan Wu. Extreme Learning Machine With Subnetwork Hidden Nodes for Regression and Classification. IEEE Transactions on Cybernetics, 46(12):2885-2898, 2016. https://doi.org/10.1109/TCYB.2015.2492468.

[11] Nieves Castro-González, Froilán M. Dopico, and Juan M. Molera. Multiplicative perturbation theory of the Moore–Penrose inverse and the least squares problem. Linear Algebra and its Applications, 503:1-25, 2016. https://doi.org/10.1016/j.laa.2016.03.027.

[12] Tao Ma, Fen Wang, Jianjun Cheng, Yang Yu, and Xiaoyun Chen. A Hybrid Spectral Clustering and Deep Neural Network Ensemble Algorithm for Intrusion Detection in Sensor Networks. Sensors, 16(10):1701, 2016. https://doi.org/10.3390/s16101701.

[13] Preeti Aggarwal, and Sudhir Kumar Sharma. Analysis of KDD Dataset Attributes - Class wise for Intrusion Detection. Procedia Computer Science, 57:842-851, 2015. https://doi.org/10.1016/j.procs.2015.07.490.

[14] Oluwafemi A. Sarumi, Adebayo O. Adetunmbi, and Fadekemi A. Adetoye. Discovering Computer Networks Intrusion using Data Analytics and Machine Intelligence. Scientific African, 9:e00500, 2020. https://doi.org/10.1016/j.sciaf.2020.e0.

[15] Nour Moustafa, and Jill Slay. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). NY, Australia, 2015. https://doi.org/10.1109/MilCIS.2015.7348942.

[16] Aniruddha Dey, and Shiladitya Chowdhury. Probabilistic weighted induced multi-class support vector machines for face recognition. Informatica, 44(4):459-467, 2020. https://doi.org/10.31449/inf.v44i4.3142.

[17] Lev A. Kazakovtsev, and Ivan Rozhnov. Application of algorithms with variable greedy heuristics for k-medoids problems. Informatica, 44(1):55-61, 2020. https://doi.org/10.31449/inf.v44i1.2737.

[18] Preeti Mishra, Vijay Varadharajan, Uday Tupakula, and Emmanuel S. Pilli. A Detailed Investigation and Analysis of Using Machine Learning Techniques for Intrusion Detection. IEEE Communications Surveys & Tutorials, 21:686-728, 2018. https://doi.org/10.1109/COMST.2018.2847722.

[19] Soo-Yeon Ji, Bong-Keun Jeong, Seonho Choi, and Dong Hyun Jeong. A multi-level intrusion detection method for abnormal network behaviors. Journal of Network and Computer Applications, 62(Feb.):9-17, 2016. https://doi.org/10.1016/j.jnca.2015.12.004.

[20] Richard Zuech, Taghi M Khoshgoftaar, and Randall Wald. Intrusion detection and Big Heterogeneous Data: a Survey. Journal of Big Data, 2(1):3, 2015. https://doi.org/10.1186/s40537-015-0013-4.

[21] Daniel S. Kermany, Michael Goldbaum, Wenjia Cai, Carolina C.S. Valentim, Huiying Liang, Sally L. Baxter, Alex McKeown, Ge Yang, Xiaokang Wu, Fangbing Yan, Justin Dong, Made K. Prasadha, Jacqueline Pei, Magdalene Y.L. Ting, Jie Zhu, Christina Li, Sierra Hewett, Jason Dong, Ian Ziyar, Alexander Shi, Runze Zhang, Lianghong Zheng, Rui Hou, William Shi, Xin Fu, Yaou Duan, Viet A.N. Huu, Cindy Wen, Edward D. Zhang, Charlotte L. Zhang, Oulan Li, Xiaobo Wang, Michael A. Singer, Xiaodong Sun, Jie Xu, Ali Tafreshi, M. Anthony Lewis, Huimin Xia, and Kang Zhang. Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning. Cell, 172(5):1122-1131.e9, 2018. https://doi.org/10.1016/j.cell.2018.02.010.

[22] Kuniaki Noda, Yuki Yamaguchi, Kazuhiro Nakadai, Hiroshi G. Okuno, and Tetsuya Ogata. Audio-visual speech recognition using deep learning. Applied Intelligence, 42:722-737, 2015. https://doi.org/10.1007/s10489-014-0629-7.

[23] Zhang Sai, Hu Hailin, Jiang Tao, Zhang Lei, and Zeng Jianyang. TITER: predicting translation initiation sites by deep learning. Bioinformatics, (14):i234-i242, 2017. https://doi.org/10.1093/bioinformatics/btx247.