ISBN: 978-981-18-3959-7
2022 the 12th International Workshop on Computer Science and Engineering (WCSE 2022)
doi: 10.18178/wcse.2022.06.045

Research on Network Security Situational Awareness Based on Convolutional Neural Network

Jiashuai Hu [1], Chaobin Wang [2 +] , Zifei Xiao [1] and Linfeng Wu [3]

[1] School of Computer Science, China West Normal University, Nanchong, China

[2] Educational Information Technology Center, China West Normal University, Nanchong, China

[3] School of Electronic Information Engineering, China West Normal University, Nanchong, China

Abstract. With the continuous development of information technology, networks are becoming larger and larger, and network security is also being severely tested. At present, traditional measures to cope with network security problems, such as installing intrusion detection systems, antivirus software and firewall systems, can protect existing network security problems, but cannot predict upcoming security problems. In response to this problem, this paper optimizes the situational awareness and prediction method based on convolutional neural network. Based on experiments with the KDDCup-99 dataset, and a comparison with traditional prediction methods, the accuracy of the prediction results is verified and the network security prediction problem is effectively solved.

Keywords: network security, situation prediction, situation assessment, convolutional neural network

1. Introduction

The American scientist Endsley[1] proposed the concept of situational awareness, and defined situational awareness as understanding and perceiving elements in the environment in a given space and time, and understanding the meaning of elements and predicting element states in the future. The first scientist to address situational awareness in network security was T. Bass[2] , who proposed an intrusion detection framework and applied it to later intrusion detection systems. Wright E[3]integrated Bayesian technology with situational awareness to provide technical support for situational awareness. Shanti[4] applied pattern recognition technology to situation awareness and used it for dynamic evaluation. Adenusi[5] applied artificial intelligence to situation awareness and used neural network models to monitor the network. In the whole process of situation awareness, situation prediction is the most important step. Currently, the most commonly used methods for situation prediction are based on the gray theory model, time series analysis and deep learning. Models based on gray theory are characterized by high accuracy and low workload. The most commonly used models are GM (1,1) and GM (1,N). The principle of the time series analysis method is to assume that the situation O_i at a certain time point has a relationship with some situations O_{i-n}, O_{i-n+1}...O_{i-2}, O_{i-1} in the previous time period of this situation, and then predict the situation at the next time according to the change law of the situation in the current time period. The method based Deep Learning is widely used in the field of situation awareness. Dong Hai[6] used the fuzzy analytic hierarchy process to divide the situation value of network security, and obtained the situation prediction method based on the radial basis function neural network. He Mengyi[7] proposed an attack detection model combining batch normalization and deep neural network to optimize model output. Shi Chen[8] proposed a quantitative formula for nominal secondary indicators to classify situation elements, and a support vector machine model for situation assessment, and an Elman neural network model for situation prediction. Based on the above research results, this paper optimizes a situation awareness method based on convolutional neural network.

[+] Corresponding author: Chaobin Wang. Tel.:13990777368
E-mail address: *cbwang@cwnu.edu.cn*

2. Principle and Design of Convolutional Neural Network

2.1. The Principle of Convolutional Neural Network

The convolutional neural network is a feedforward neural network with a deep structure. It is one of the classic Deep Learning algorithms and is widely used in computer vision and natural language processing[9]. A convolutional neural network (CNN) usually includes an input layer, convolutional layers, pooling layers, full connection layers, and an output layer. The network model with only one convolutional layer, one pooling layer and one full connection layer is shown in Figure 1.

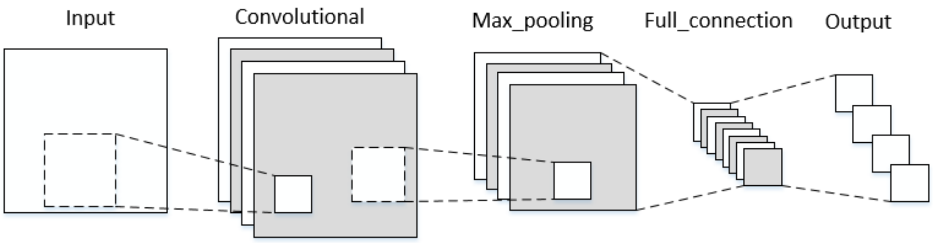


Fig. 1: Convolutional neural network model

The layers are described as follows:

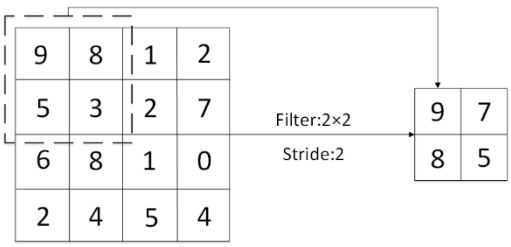
Input layer: The function is to receive the processed data and input the data to the convolution layer.

Convolution layer: The function is to use the convolution kernel to perform the convolution operation on the input data. Each element of the convolution kernel has its corresponding weight and bias. After convolution, the activation function extracts the features[10]. Assuming that the feature map has the same length and width, the formula for the convolution operation is:

$$Z^{l+1}(m,n) = \sum_{k=1}^{k_l} \sum_{x=1}^{f} \sum_{y=1}^{f} \left[Z_k^l\,(gm+x, gn+y) w_k^{l+1}(x,y) \right] + b$$
$$(m,n) \in \{0，1，\ldots L_{l+1}\} \; L_{l+1} = \left\lfloor \frac{L_l + 2P - f}{g} + 1 \right\rfloor \qquad (1)$$

In formula (1), Z^l is the convolution input of the l+1th layer, Z^{l+1} represents the convolution output of the l+1th layer, k represents the number of convolution kernels, L_{l+1} and L_l represent the size of the l+1th and lth layers, f represents the size of the convolution kernel, g represents the stride, P represents the number of padding layers.

Pooling layer: The function is to reduce the dimension and shrink the feature map after convolution, generally using the maximum pooling method. The pooling operation can effectively extract important features and reduce the operation parameters, and there is no activation function in the pooling layer[11]. In the maximum pooling calculation method, the filter is used to extract the maximum value of each region and create a new feature map, as shown in Figure 2.



9	8	1	2
5	3	2	7
6	8	1	0
2	4	5	4

9	7
8	5

Fig. 2: Example of max pooling

Figure 2 shows the process of max pooling. The size of the original feature map is 4×4, the size of the filter is 2×2, and the step size is 2. The filter extracts the maximum value of each region, respectively 9, 7, 8, 5. These maxima form a new feature map of size 2×2.

Full connection layer: The function is to connect all the features, calculate the activation value by the activation function, and send the activation value to the classifier[12].

Output layer: The function is to receive each activation value from the full connection layer, and use the classifier to classify and output the features[13].

2.2. The Design of Convolutional Neural Networks

The network structure used in this paper is shown in Figure 3:

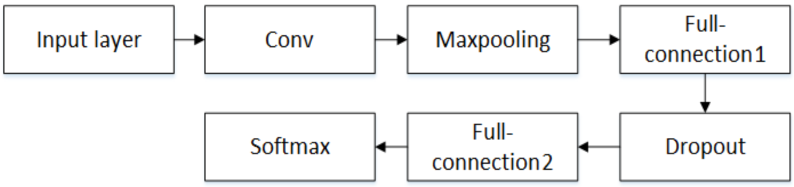


Fig. 3: The convolutional network structure used in this paper

The parameters of each layer are initially set as follows:

Convolution layer: The size of the convolution kernel generally follows the principle of small convolution and odd number[14]. We set its size to 3 ×3, the number to 32, and the step size to 1.

Pooling layer: The pooling layer chooses the maximum pooling method, the sliding window size is 2 × 2, and the step size is 2.

Full connection layer: The first full connection layer dimension is set to 1 × 256, and the second to 1 × 128. Two full connection layers are set to integrate features and reduce dimensionality.

Dropout layer: The convolutional network tends to overfit during training, which affects the experimental effect. To mitigate this problem, add a dropout layer between the two full connection layers. The value is set to 0.25.

Softmax layer: In this experiment, a total of 40 outcomes are predicted, so the dimension is set to 1 ×40.

3. Experiment and Analysis

3.1. Introduction to the Experimental Environment and Dataset

Hardware environment of this experiment: The CPU is Inter (R) Core (TM) i7-6700, 3.40GHz. Software environment: PyCharmCommunityEdition2021.2.1, Anaconda3, Python3.8, Sklearn, Pandas, etc. and the keras deep learning library in the Tensorflow framework. The dataset used for the experiment is the KDDCup-99 dataset, in which each piece of data contains 41 attributes and 1 label. The labels are divided into normal types and 4 abnormal types. These 4 abnormal types contain 39 specific abnormal identifiers. Since the data is too large, the experiment uses 10% of it as the training set, with a total of 494,021 data, and a total of 311,029 data in the test set[15].

3.2. Data Preprocessing

Numeric replacement text: The protocol type, network service type, network connection status, and attack type in the original data are all text types. All of these texts are converted to numeric values so that they can be entered into the neural network after normalization. For example, “TCP” in the protocol type is replaced with “0”, “UDP” is replaced with “1”, and “ICMP” is replaced with “2”.

Numerical normalization: Since the distance between some values in the data set is too large, the value with the highest value is normalized. After normalization, the value is in the range [0,1], which solves the problem that the data is not in the same dimension. The formula for the maximum normalization is:

$$X = \frac{X_i - X_{min}}{X_{max} - X_{min}} \tag{2}$$

In formula (2), Xmax and Xmin are the maximum and minimum values of the ith dimension attribute, Xi is the current value, and X is the normalized value.

Label one-hot encoding: The principle of one-hot encoding is to use N registers to encode N states, and these registers have only a single state. One-hot encoding converts the discrete features in the dataset into continuous values, that can be better accepted by the neural network model. The experiment is a multi-classification problem, which is divided into 40 subclasses at the end. Therefore, the one-hot encoding of the label should be represented by 40 status bits. The register code corresponding to the classification is 1, and the remaining codes are 0[16].

3.3. Selection of Experimental Parameters

The experimental activation function chooses the nonlinear function Relu. The essence of this experiment is a multi-classification problem, the loss function chooses the categorical_crossentropy, and the evaluation standard chooses the accuracy[17]. Completion of the initial construction of the network model, input the preprocessed training data into the network model. The training accuracy is shown in Figure 4, and the training loss value is shown in Figure 5:

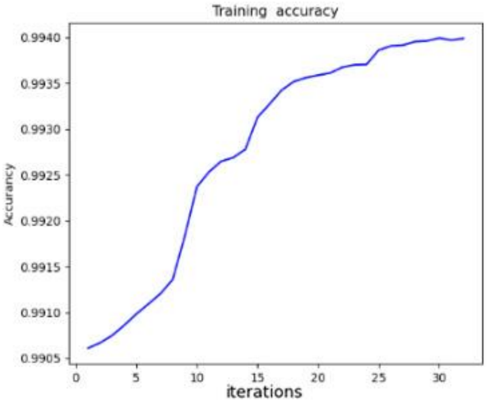


Fig. 4: Training accuracy

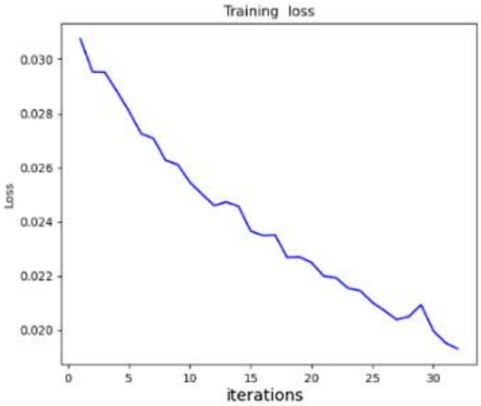


Fig. 5: Training loss

From Figure 4 and Figure 5, we can see that as the number of iterations increases, the accuracy increases and the loss decreases, indicating that the network training has achieved good results. Parameters that affect the experimental results are also the number of iterations (iterations), the number of training samples per batch (batch-size), and the choice of optimization function. Figure 4 shows that the accuracy curve gradually converges when the number of iterations exceeds 30. Therefore, set it to 10, 20, 30, 40, and 50. The batch size is generally chosen to be 2^n, so set it to 32, 64, 128, and 256. After setting the iterations and batch size, send the test data to the trained network model. The effects of the different iterations and batch sizes on test accuracy are shown in Table 1:

Table 1: Test results of different iterations and batch-size

	batch-size			
Iterations	32	64	128	256
10	0.9023	0.9136	0.9169	0.9166
20	0.9056	0.9130	0.9189	0.9170
30	0.9169	0.9184	0.9219	0.9205
40	0.9153	0.9167	0.9180	0.9179
50	0.9122	0.9143	0.9166	0.9160

From Table 1, we can see that the experimental effect is best when iterations=30 and batch size=128. In multi-classification problems, the optimization functions SGD, Adam, Adadelta and Rmsprop are usually used. Under the conditions of iterations=30 and batch size=128, respectively, using SGD, Adam, Adadelta and Rmsprop to conduct experiments, the test accuracy of the different optimization functions is shown in Table 2:

Table 2: Test accuracy of different optimization functions

Function	SGD	Adam	Adadelta	Rmsprop
Accuracy	0.917	0.9206	0.913	0.9147

From Table 2, we can see that the result is most ideal when using the Adam optimization function. Finally, this paper chooses the number of iterations of the network to be 30, the batch size to be 128, and the optimization function to be Adam.

3.4. Situation Assessment and Forecast

For the situation assessment experiment, if the data set contains N data, each network data corresponds to a separate label. Number the network data as D(0), D(1), D(2)... D(n-2), D(n-1). The labels corresponding to these network connections are L(0), L(1), L(2)...L(N-2), L(N-1). For the situation prediction experiment, the experimental process is roughly the same as for the situation assessment experiment, but the purpose of the situation prediction is to predict the possible attacks on the network in the future, so the label value corresponding to each network data in the dataset should be the label of the next network data[18]. Number the network data as D(0), D(1), D(2)...D(n-2). The labels corresponding to these network connections are L(1), L(2)...L(N-2), L(N-1). When sending the processed data to the trained CNN network model, the accuracy of situation assessment is 0.9227, and the accuracy of situation prediction is 0.9116. In order to verify that the CNN model in this paper has higher accuracy than other traditional algorithms in situation assessment and situation prediction, under the same data set, using the CNN model optimized in this paper, K-means, SVM, RF and the improved LSTM network proposed in the literature[19] to experiment. The experimental results are shown in Table 3:

Table 3: Experimental accuracy of different algorithms

Method	CNN	K-means	RF	Improve LSTM	SVM
Evaluate Accuracy	0.9227	0.7762	0.8761	0.9056	0.8846
Prediction Accuracy	0.9116	0.7690	0.8579	0.8975	0.8762

From Table 3, we can see that the accuracy of situation prediction in each method is slightly lower than that of situation assessment, but the CNN model optimized in this paper has higher accuracy than other methods in both evaluation and prediction, which reflects the advantages of the model. To see the effect of situation prediction more intuitively, the situation data are quantified according to the situation value quantification method in the literature [20], and the situation value is normalized within [0,1]. Selecting 15 data samples from the situation prediction data set according to the law of situation value from low to high to low, and comparing the real situation value of the data sample with the prediction results of CNN, improved LSTM and SVM with better experimental results. The result is shown in Figure 6:

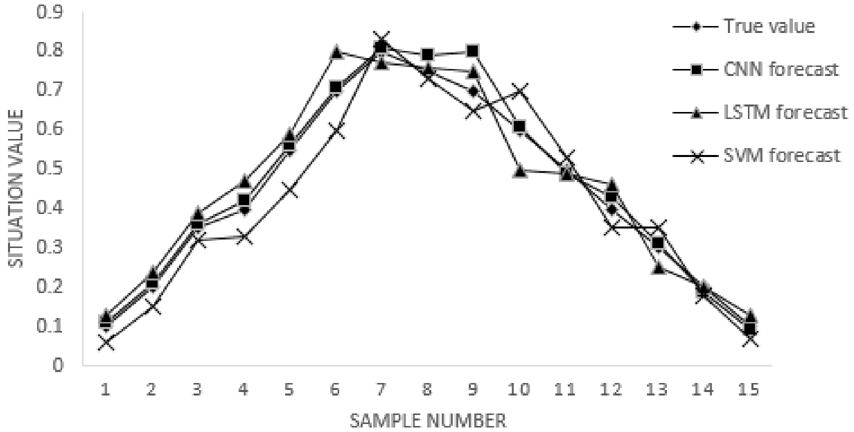


Fig 6: Comparison of the predicted value of different methods with the actual value

From Figure 6, we can see that the improved LSTM model is far from the real situation for samples No. 6, No. 7, No. 9, and No. 10, and the SVM model is far from the real situation for samples No. 4, No. 5, No. 6, No. 9, and No. 10. However, the CNN model is far from the real situation only for samples No. 8 and No. 9, and the number of matches is higher than the other two types of models. It has proved that the CNN model is more accurate than the other two models.

To further verify the accuracy of the experimental model, three error indicators are used to evaluate the three methods of situation prediction, namely the mean absolute error (MAE), and the formula is:

$$\mathrm{MAE} = \frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}} |\,\mathrm{real} - \mathrm{pre}\,| \tag{3}$$

Root Mean Square Error (RMSE), the formula is:

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}} (\,\mathrm{real} - \mathrm{pre}\,)^2} \tag{4}$$

Mean Absolute Percent Error (MAPE), the formula is:

$$\mathrm{MAPE} = \frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}} \left|\frac{\mathrm{real} - \mathrm{pre}}{\mathrm{real}}\right| \times 100\% \tag{5}$$

In the formulas (3), (4), and (5), real represents the value of the actual situation, N represents the number of samples, pre represents the value of the predicted situation. From the formulas, we can see that the smaller the values of MAE, RMSE and MAPE, the better the performance of the model. We put the prediction data of the three models into the calculation, and the results are shown in Table 4:

Table 4: Comparison of error indicators of predicted situation value

Error indicators	CNN	SVM	Improved LSTM
MAE	0.0200	0.0513	0.0440
RMSE	0.0306	0.0583	0.0524
MAPE	5.0982	15.2621	12.8698

Table 4 shows that the values of MAE, RMSE and MAPE calculated by the CNN model are smaller than those of the other two models. It is further proved that the CNN model is superior to the other two models in situation prediction.

4. Conclusions

This paper optimizes a method for situation assessment and prediction based on convolutional neural network. First, the KDDCUP-99 dataset is prepeocessed and the main parameters affecting the performance of the network are determined. To verify the feasibility of the method, compared the method in this paper with the traditional method, and found that the method in this paper has a higher accuracy rate in situation assessment, agrees better with the real situation value in situation value prediction, and has a lower error value in performance index evaluation. It shows that the method in this paper can evaluate the situation more accurately and predict the future situation. This will help network security officers make faster and more accurate decisions about upcoming cyber risks and better protect network security.

5. Acknowlegements

Project supported by the Talent Research Fund Project of China West Normal University (Grant No. 17YC498)

6. References

[1] ENDSLEY M. Design and Evaluation for Situation Awareness Enhancement [C]. Proceedings of the Human Factors and Ergonomics Society Annual Meeting,1988,32(1):97-101.

[2] Bass T, Gruber D. A glimpse into the future of ID[OL].URL: [2016-03-10] http://static.usenix.org/legacy/publications/login/1999-9/features/future.html.

[3] Wright E, Mahoney S, Laskey K, et al. Multi-entity Bayesian networks for situation assessment[C]// Proceedings of the Fifth International Conference on Information Fusion. FUSION 2002. (IEEE Cat.No.02EX5997). IEEE,2002:804-811.

[4] K. Shanti Swarup. Artificial neural network using pattern recognition for security assessment and analysis[J]. Neurocomputing,2007,71(4).983-998.

[5] D. Adenusi, B. K. Alese, B. M. Kuboye, et al. Development of cyber situation awareness model[C]. 2015 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (Cyber SA), 2015, 1-11.

[6] Dong Hai. Research and implementation of network security situational awareness system based on GA-RBF neural network [D]. Yinchuan: Ningxia University, 2017

[7] He Mengyi. Research on network security situational awareness technology based on neural network [D].

Mianyang: Southwest University of Science and Technology, 2020

[8] Shi Chen. Research and implementation of network security situational awareness system based on neural network [D]. Nanjing: Nanjing University of Posts and Telecommunications, 2020

[9] Li Bingzhen. Review of convolutional neural networks [J]. Computer age, 2021

[10] Gu J, Wang Z, Kuen J, Ma L,2015. Recent advances in convolutional neuralnetworks. ar Xiv preprint arXiv:1512.07108

[11] Zeiler M D, Fergus R. Visualizing and understanding convolutional networks[C]//European conference on computer vision. Springer, Cham,2014:818-833

[12] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large Scale Image Recognition[J]. Computer ence, 2014.

[13] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of International Conference on Computer Vision and Pattern Recognition, 2015.

[14] Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition,2017:4700-4708

[15] Zhu Chenfei. Research on network security situation assessment and prediction method based on Neural Network[D]. Beijing: People's Public Security University of China, 2019

[16] Ren Dezhi. Research on feature extraction technology of security situation awareness based on machine learning[D]. Chengdu: University of Electronic Science and Technology of China, 2020

[17] Zhang Xin. Research on acquisition and evaluation technology of network security situation elements[D]. Chongqing: Chongqing University of Posts and Telecommunications, 2021

[18] Liu Bin. Research on Key Technologies of network security situation assessment and prediction[D]. Chengdu: University of Electronic Science and Technology of China, 2021

[19] Li Shixuan. Research on network security situational awareness based on improved LSTM neural network [D]. Shijiazhuang: Hebei Normal University, 2020

[20] Xie Lixia, Wang Yachao. A new method of network security situational awareness [J]. Journal of Beijing University of Posts and Telecommunications, 2014, 37(5): 31-35.