Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 3, No. 3, 2023

Research on Network Intrusion Detection Based on Transformer

Gang Gan, Weiju Kong

School of Cyberspace Security, Chengdu University of Information Technology, Chengdu 610225, China.

Abstract: With the advancement of technology, the development of various industries has become inseparable from informatization. People's lives have become closely related to the network. While using the network to facilitate our lives, massive data is also generated. Traditional firewall technologies are no longer sufficient to meet current needs. Deep learning algorithms can establish complex mapping relationships between network data, and can extract hidden correlation features between data features to achieve data recognition and prediction. Therefore, this paper introduces Transformer and Bidirectional Long Short-Term Memory (BiLSTM) into the field of intrusion detection, and proposes an intrusion detection method based on the combination of Transformer-Encoder and BiLSTM (TBL). Deep Neural Networks (DNN) are used to further extract data features, and the softmax function is used to output classification results. In order to verify the effectiveness of this method, this paper trains and tests the TBL method on the NSL-KDD dataset, and verifies its feasibility and superiority.

Keywords: Intrusion Detection; Deep Learning, Transformer; Bidirectional Long Short-Term Memory.

1. Introduction

As a network defense system that can identify network attack behaviors and recognize and alert against them, Intrusion Detection System (IDS) has become increasingly important in recent years. Artificial intelligence has made significant breakthroughs in many fields, especially in data recognition and classification, where neural networks with their powerful feature extraction and learning capabilities can efficiently identify and classify massive amounts of data in networks. Researchers at home and abroad are studying the most effective learning models in different scenarios, but few have studied how to apply neural networks to network intrusion detection. Therefore, defending against network attacks is of great significance for national security, enterprise development, and personal privacy protection. Today, with the rapid development of network technology, traditional intrusion detection techniques have gradually become ineffective, and developing new network intrusion detection models has become a top priority. Building an intrusion detection system based on deep learning has both theoretical research value and practical application value. This paper proposes an intrusion detection method that combines entity embedding and Transformer, using attention mechanisms to identify and classify attack behaviors in network traffic to achieve the goal of protecting important information. The NSL-KDD dataset is used to validate the performance of the model, and the experimental results show that the model has good detection performance.

Yang L's team [1,2] proposed the application of convolutional neural networks to network intrusion detection algorithms, and the experimental results showed that the method not only has generalization but also improves the convergence speed of the model. Xiao Y's team [3,4] proposed a simplified residual network algorithm to reduce the complexity of the network and prevent the problem of model overfitting. The algorithm further simplifies the original residual algorithm by deleting one weight layer and two batch normalization layers. Although some progress has been made in preventing overfitting, the detection results for new types of network attacks are not ideal. Yu's team [5,6] proposed a BiLSTM (Bidirectional Long Short-Term Memory) intrusion detection model, which can capture key information in the feature information to complete the detection of abnormal data. Although it fully learns the temporal information of attack behavior, it did not consider the impact of historical information, but still provides a new idea for intrusion detection.

Research experiments have shown that the imbalance of a dataset has a significant impact on the final training results of a model. Imbalanced data can result in a higher proportion of minority class samples in the dataset, which can significantly affect the accuracy of the model's predictions. Some scholars have begun researching the issue of data balancing. The Bedi [7] team used Siamese neural networks (Siamese-NN) to address the issue of class imbalance in datasets. While this method has made some progress in detecting minority class anomalies, its precision still needs improvement. Li Chuan [8] proposed using Generative Adversarial Networks (GANs) to expand the number of minority class samples in the data, comparing the expanded data with the original data, and subsequently conducting comparative experiments on the original and expanded data using CNNs. The experimental results showed that GANs were effective in expanding the dataset. The Fu [9] team used an adaptive synthetic sampling algorithm to expand the number of minority class samples to address the issue of data imbalance, but this method produced a large amount of noise data due to the influence of surrounding data, ultimately negatively impacting the model's performance.

2. Transformer_BiLSTM

2.1. Model Architecture

The model in this paper first preprocesses the initial data and inputs the processed data into the Transformer module to establish connections between different features, and extracts richer feature information through multi-head attention. Then, the data is input into a Bi-LSTM neural network to obtain the connection between the previous and next features to retain

its temporal information. Finally, features are further extracted through DNN and classified and recognized using a SoftMax classifier to obtain the results.

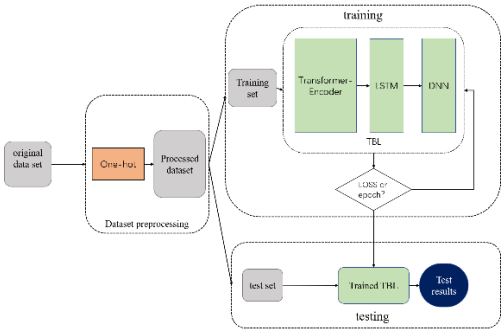

Figure 1. TBL intrusion detection model

2.2. Transformer-Encoder

The original Transformer model is divided into two parts: the encoder and the decoder. In this chapter, we only used the encoder module of the Transformer model and made some adjustments. The encoder is composed of multiple stacked Encoders, each of which includes a multi-head attention mechanism, a feedforward neural network, and a residual network. To increase training speed and save time, the attention mechanism uses dot-product attention that can be computed in parallel, with the specific formula shown in Equation (1).

$$\text{Attention}(Q,K,V)=\text{softmax}(\frac{Q\cdot K^T}{\sqrt{d_k}})\cdot V \quad (1)$$

The feedforward neural network has only one hidden layer, which is a perceptron with the same input and output dimensions. Since a single hidden-layer network has weak non-linear mapping ability, and considering the balance between computational complexity and mapping ability, the number of hidden layer neurons is set to twice the number of input layer neurons in this paper [10,11].

The activation function ReLU is used. The structure of the entire Transformer-Encoder module is shown in Figure 2.

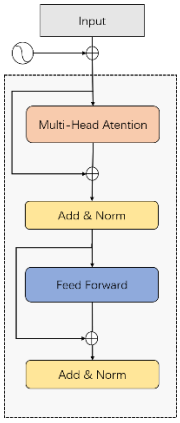

Figure 2. Transfirmer-Encoder

2.3. BiLSTM

The Long Short-Term Memory (LSTM) neural network is a type of recurrent neural network (RNN) that solves the problem of long-range information loss in long sequences. It is used for processing time-series information and addresses the issues of gradient explosion and vanishing in RNN structures. It is capable of memorizing valuable information while discarding redundant memories [12]. The Bidirectional LSTM consists of a forward LSTM and a backward LSTM, which combine to contain all the information from both directions. Its structure is shown in Figure 3.

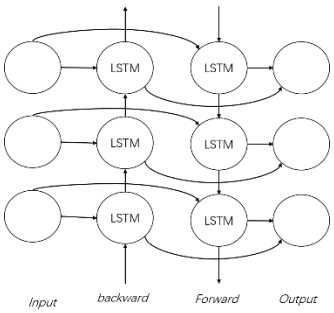

Figure 3. BiLSTM

The input layer, Input, takes the input data and feeds it into both the forward network, Forward, and the backward network, Backward [13,14]. The outputs from these networks are concatenated and represented as follows:

$$h_i=[\overrightarrow{h_i}, \overleftarrow{h_i}] \quad (2)$$

Here, hi represents the final result obtained by concatenating the outputs of the forward and backward networks for the i-th input, where i ranges from 1 to n, where n is the total number of input data.

2.4. DNN

DNNs are generally classified based on the different layers, which can be divided into three categories: input layer, hidden layer, and output layer. Typically, the first layer is the input layer, the last layer is the output layer, and all layers in between are hidden layers. The network structure of DNN is shown in Figure 4.

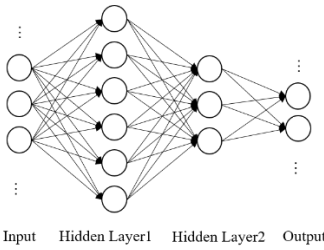

Figure 4. DNN

The general calculation formula (3) for DNN is as follows:

$$f(x)=\sigma(WX+b) \quad (3)$$

3. Results and Analysis

3.1. Environment

The intrusion detection model based on TBL uses a Windows operating system, an Intel(R) Core (TM) i7-11700H CPU @ 2.50GHz processor, and 16GB of memory. The model training is accelerated using the NIVDIA GeForce RTX 3060Ti GPU, and the programming tools used are Pytorch, Cuda11.0, and Python 3.6.

3.2. Dataset

The dataset used in this article's experiments is the NSL-KDD dataset, which is a subset of the classic KDD99 dataset with an additional feature. To avoid the classifier being biased towards frequent records and to ensure accurate detection, the NSL-KDD dataset removes a large amount of redundant and duplicated data from the KDD99 dataset and reasonably splits it into training and testing sets.

Each record in the NSL-KDD dataset contains 43 features, where the first 41 features represent the traffic itself, the 42nd feature is the label indicating whether the record is normal or an attack behavior, and the 43rd feature is an additional feature added by NSL-KDD on the KDD99 dataset, representing the difficulty score of detecting the record.

The dataset contains a total of 5 classes of data, including 4 types of attack and 1 type of normal data. The four types of attack are: Dos, Probe, U2R, R2L. These 4 types of attacks are further divided into 38 attack methods, with 22 types of attacks included in the training set and the remaining attack types included in the testing set. The generalization ability of the model is evaluated by testing whether it can detect unknown attack types in the testing set. The division of attack types in the training and testing sets is shown in Table 1.

Table 1. Division of Attack Types in Subsets

Type	Attack types in the training set	Attack types in the test set
Dos	Back, land, Neptune, pod, smurf, teardrop	Apache2, mailbomp, processtable, udpstorm, worm
Probe	Ipsweep, nmap, portsweep, satan	Mscan, saint
U2R	Buffer_overflow, loadmodule, perl, rootkit	Ps, sqlattack, xterm
R2L	ftp_write, guess_passwd, imap, multihop, phf, spy, warezmaster	Named, Httpummel, sendmail, Snmpgetattac, snmpguess, xlock, snaoop

The NSL-KDD dataset is divided into four subsets: KDDTest+, KDDTest-21, KDDTrain+, and KDDTrain+20%. KDDTest-21 is a subset of KDDTest+ that removes data with the 43rd feature equal to 21, which represents easily detectable intrusion data. Therefore, the KDDTest-21 dataset consists of difficult-to-detect traffic data. KDDTrain+20% is a subset of KDDTrain+ with only 20% of its data. Table 2 shows the number of instances and the proportion of each data type in each subset of the NSL-KDD dataset.

Table 2. Quantity and proportion of various types of data in the data set

Dataset	Normal	Dos	Probe	U2R	R2L
KDDTrain	13449	9834	2289	11	209
KDDTrain+20%	67343	45927	11656	52	995
KDDTest++	9711	7460	2421	67	2885
KDDTest-21	2152	4342	2402	200	2754

3.3. Performance Metrics

In the field of intrusion detection, normal traffic samples are usually referred to as negative samples, while samples of attack types are referred to as positive samples. There are four possible outcomes for all samples that are detected: (1) True Positive (TP): Actual attack samples that are correctly identified as attack samples. (2) False Positive (FP): Actual normal samples that are incorrectly identified as attack samples, also known as false alarms. (3) True Negative (TN): Actual normal samples that are correctly identified as normal samples. (4) False Negative (FN): Actual attack samples that are incorrectly identified as normal samples, also known as misses.

There are four evaluation metrics for intrusion detection, as follows:

(1) Accuracy (ACC) represents the proportion of successfully detected samples to the total samples. A higher value indicates a better model. Its calculation formula is as follows:

$$\text{Accuracy(ACC)}=\frac{TP+TN}{TP+TN+FP+FN} \tag{4}$$

(2) Precision represents the proportion of actual attack samples among all samples that are detected as attack by the model. A higher value indicates better performance in detecting attack behaviors. Its formula is:

$$\text{precision}=\frac{TP}{TP+FP} \tag{5}$$

(3) Recall represents the proportion of samples identified as attack type among all attack samples in the dataset. A higher value indicates better performance of the model. Its formula is:

$$\text{Recall}=\frac{TP}{TP+FN} \tag{6}$$

(4) The F1-Score is a comprehensive evaluation of precision and recall, with a maximum value of 1 and a minimum value of 0. A higher value indicates a better model performance. Its formula is:

$$F_1\text{Score}=\frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \tag{7}$$

3.4. Parameter Settings

The parameter settings for the intrusion detection model based on TBL are shown in Table 3:

Table 3. Model parameters

Parameters	Value
iterations	100
batch size	2000
Attention Heads	4
Feedforward Hidden Nodes	64
BILSTM hidden nodes	64
learning rate	0.0001
loss rate	0.3

3.5. Analysis of Results

The detection accuracy of the intrusion detection models based on the commonly used machine learning algorithms SVM and Random Forest (RF), as well as the commonly used deep learning network models Transformer and LSTM, were compared in experimental tests, as shown in Figure 5.

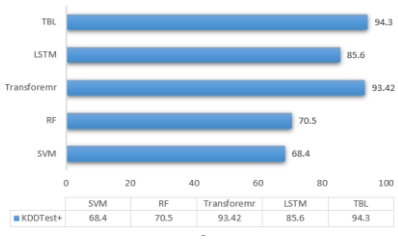

Figure 5. Loss value

KDDTest+ dataset, and it can be seen from the comparison in the figure that the TBL method designed in this chapter has a significant advantage over other methods, with an accuracy of 94.3%. Therefore, the methods discussed in this chapter are feasible.

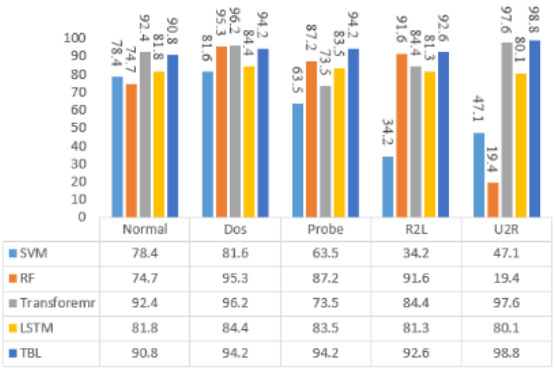

Figure 6. Accuracy

Figure 6 compares the five-class precision of different models on the KDDTest+ dataset. The data from the graph shows that the TBL-based intrusion detection method designed in this chapter has certain advantages in detecting the attack types Probe, R2L, and U2R, with precision rates of 94.2%, 92.6%, and 98.8%, respectively. However, the performance of this method in detecting Normal and Dos types of data is not as good as that of the Transformer-based intrusion detection method. The method designed in this chapter has a detection effect on Normal data that is lower than the best by 1.6% and a detection effect on Dos type data that is lower than the best by 2.0%. Overall, the method designed in this chapter is still effective.

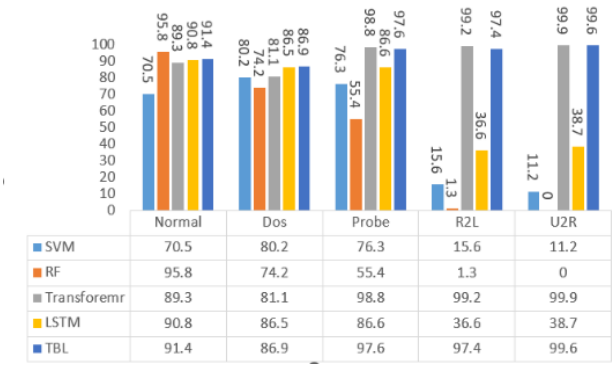

Figure 7. Recall

Figure 7 shows the recall rate of different methods in detecting the KDDTest+ dataset, which mainly expresses the probability that each type of data can be correctly distinguished in different types of data. From the figure, it can be seen that for Normal type data, the recall rate of the RF method is the highest at 95.8%, but the RF model's recall rate for Probe, R2L, and U2R type data is not good. The TBL-based intrusion detection method designed in this chapter has improved the recall rate on Normal and Dos type data by 2.1% and 5.8%, respectively, compared to the Transformer-based intrusion detection method. However, the recall rate of the TBL model in detecting Probe, R2L, and U2R type data is slightly lower than that of the Transformer model. The TBL model proposed in this chapter is an attempt in the field of intrusion detection. Comparing the precision and recall rates can show that this model has certain effects and provides a new direction for intrusion detection based on Transformer.

To provide a more intuitive comparison of the performance of different models, the F1 values were compared to evaluate the overall performance of the models. The F1 values of each model are shown in the following Figure 8.

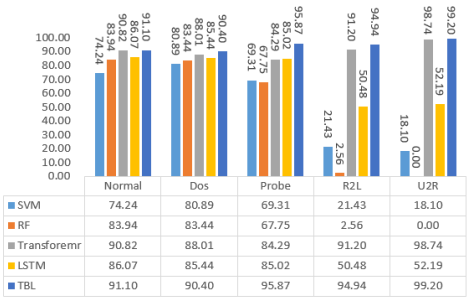

Figure 8. F1

The F1 score is generally an important indicator for overall performance comparison of models. The F1 score comparison of different models for each data type is shown in Figure 8. From the figure, it can be seen that the TBL model proposed in this paper has an overall advantage and achieved 91.1%, 90.4%, 95.87%, 94.94%, and 99.2% for the five different types of data, respectively. This indicates that the TBL-based intrusion detection method proposed in this chapter is worth studying.

4. Conclusion

With the advancement of technology, the development of various industries has become inseparable from information technology. People's lives have become closely related to the Internet, and while the network facilitates our lives, it also generates massive amounts of data, including sensitive data involving personal privacy. Once such data is leaked, the resulting losses are incalculable. Therefore, network security issues have received increasing attention. With the complexity and diversity of attack methods, traditional firewall technologies can no longer meet the current security needs. In contrast, intrusion detection methods based on deep learning can automatically discover hidden associations in traffic data, solve the problem of manual feature selection in traditional methods, and can better identify unknown attack types and adapt to the heterogeneous network traffic data and rapidly changing attack patterns, which can meet the needs of protecting against the endless stream of attack methods in the current network environment. Therefore, combining intrusion detection systems with deep learning has become a new goal for researchers in the intrusion detection field. Based on an understanding of existing deep learning-based intrusion detection methods, particularly the problems of using a single model in intrusion detection methods, this paper analyzes and proposes a novel composite model-based intrusion detection method and applies it to the intrusion detection field, verifying its feasibility and effectiveness through experiments.

References

[1] Yang L, Shami A. A transfer learning and optimized CNN based intrusion detection system for internet of vehicles[J]. arXiv preprint arXiv:2201.11812, 2022.

[2] Xiao Y, Xiao X. An intrusion detection system based on a simplified residual network[J]. Information,2019, 10(11): 356.

[3] YU Y, LIU G, YAN H, et al. Attention-based BiLSTM model for anomalous HTTP traffic detection[C]//2018 15th International Conference on Service Systems and Service Management, 2018: 1-6.

[4] BEDI P, GUPTA N, JINDAL V. Siam-IDS: handling class imbalance problem in intrusion detection systems using siamese neural network[J].Procedia Computer Science, 2020, 171: 780-789.

[5] Li Chuan. Research and Implementation of Intrusion Detection Based on Generative Adversarial Networks [D]. North China Electric Power University (Beijing),2022.DOI:10.27140 /d.cnki.ghbbu. 2022.000269.

[6] FU Y, DU Y, CAO Z, et al. A deep learning model for network intrusion detection with imbalanced data[J]. Electronics, 2022, 11(6): 898.

[7] Staudemeyer R C .Applying long short-term memory recurrent neural networks to intrusion detection[J]. South African Computer Journal, 2015,(1):136-154.

[8] Volodymyr Mnih,Nicolas Heess,Alex Graves,Koray Kavukcuoglu. Recurrent Models of Visual Attention. arXiv. preprint arXiv:1406.6247v1.2014.

[9] Mhaskar H N, Micchelli C A. How to choose an activation function[J].Advances in Neural Information Processing Systems,1994: 319-326.

[10] He K, Zhang X, Ren S , et al. Identity Mappings in Deep Residual Networks[C]// European Conference on Computer Vision. Springer, Cham, 2016.

[11] Luong M T, Pham H, Manning C D. Effective approaches to attention-based neural machine translation[J]. arXiv preprint arXiv:1508.04025, 2015.

[12] Mnih V, Heess N, Graves A. Recurrent models of visual attention[J]. Advances in neural information processing systems, 2014, 27.

[13] Olanow C W , Koller W C . An algorithm (decision tree) for the management of Parkinson's disease: Treatment guidelines[J]. Neurology, 1998, 50(3 Suppl 3):S1.

[14] Alrawashdeh K , Purdy C . Toward an Online Anomaly Intrusion Detection Systen Based on Deep Learning[C].//2016 15th IEEE International Conference on Machine Learning and Applications(ICMLA).IEEE,2016.