A Fusion Network Intrusion Detection Model Based on CNN-LSTM

Wenhao Jiang[1,a], Zheng Li[2.b]

[1]College of Intelligent Information Engineering, Chongqing Aerospace Polytechnic, Chongqing, China

[2]Mobile Communication Engineering Research Center, Chongqing University of Posts and Telecommunications, Chongqing, China

[a]yljdp@yeah.net, [b]lizheng@cqupt.edu.cn

Abstract. The explosive growth of network and Internet traffic has brought unprecedented challenges. Existing network intrusion detection systems are unable to cope with unknown network attacks and achieve real-time network response. To solve this problem, this paper proposes a hybrid deep learning model that integrates convolutional neural network (CNN) and improved long short-term memory network (LSTM). Convolutional neural network is used to obtain the spatial features of network intrusion detection data, and improved long short-term memory network is used to obtain the long-term and short- term temporal features of network intrusion detection data, thereby retaining the spatial and temporal dependencies of the data. The proposed hybrid deep learning model is applied to publicly available datasets to test its performance, and satisfactory results are obtained, which verifies that the model can efficiently detect network intrusions.

Keywords: Network Intrusion Detection, Convolutional Neural Network, Long Short-Term Memory Network.

1. Introduction

Network Intrusion Detection[1] has been an important research direction in the field of network security in 1986. However, with the advent of the big data era, traditional network intrusion detection methods face challenges such as huge data volumes, complex and changing attacks, and the need for real-time response. When network intrusion detection uses machine learning methods, this method relies on shallow machine learning and is difficult to extract complex intrusion data features, resulting in low network intrusion identification accuracy and insufficient real-time performance.

In order to overcome the limitations of network intrusion detection as mentioned above, this paper proposes a network intrusion detection model that combines CNN [2] and LSTM [3]. The performance of the fusion deep learning model is tested on the public UNSW-NB15[4] dataset. The experimental results show that the performance of the model is better than other existing models.

1.1 Main Contributions

The main contributions of this paper are as follows:

Utilize the weight sharing characteristics of CNN to efficiently extract the spatial features of Network Intrusion data, thereby improving the processing speed of Network Intrusion Detection.

Use improved LSTM to learn the long-term and short-term temporal dependencies between Network Intrusion data, thereby improving the ability to identify complex Network Intrusions.

We use drop-connect[5] to regularize the weight matrix of the data to avoid overfitting. We optimize the hyperparameters of this model through repeated trials and evaluate the performance of this model on the public UNSW-NB15 dataset.

1.2 Article Structure

The rest of the article is structured as follows:

The second part comprehensively reviews related research, introduces deep learning models, and focuses on the application of deep learning models in the field of network intrusion detection.

The third part proposes a deep learning model that integrates CNN and LSTM, and elaborates on the specific implementation process of the model.

The fourth section presents the experimental process and experimental results of this model and evaluates the performance of the proposed model.

The fifth section summarizes the model proposed in this paper and discusses future research directions and further improvement measures.

2. Related work

In recent years, there has been a trend of using deep learning technology to handle various network intrusion detection. Commonly used deep learning technologies include Deep Belief Networks(DBN), Deep Residual Networks(ResNets) and Generative Adversarial Networks(GANs).

2.1 Deep Belief Networks (DBN)

Deep Belief Networks (DBN)[6] is a Generative Graph model stacked by multiple Restricted Boltzmann Machines (RBMs). It is used in network intrusion detection systems (Intrusion Detection). Systems, IDS) has shown significant potential. DBN can perform feature extraction through unsupervised learning and can be fine-tuned through the back-propagation algorithm to improve the accuracy of intrusion detection of abnormal behaviors in network traffic.

Deep Residual Networks (ResNets)

Deep Residual Networks (ResNets)[7] solve the gradient disappearance and gradient explosion problems in deep network training by introducing a residual learning framework. The core idea of ResNets is to allow signals in the network to bypass one or more layers and pass directly by adding skip connections, so that the network can effectively train very deep hierarchical structures. In Network Intrusion Detection Systems the residual block structure and skip connections of ResNets enable the network to maintain or even improve performance while increasing depth , rather than experiencing degradation. This feature is very important for Network Intrusion Detection. This is especially important because data in the cybersecurity domain are often of high dimensionality and complexity, requiring Deep Networks to capture richer feature representations.

2.2 Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs)[8] are powerful generative models that generate high-quality data through adversarial training between a generator and a discriminator. GANs have shown great potential in Network Intrusion Detection Systems, and their applications often involve generating realistic network traffic data, including both normal traffic data and various attack data. In this way, GANs can help enhance the dataset of intrusion detection systems, especially in the face of data imbalance or lack of sufficient labeled data. The generated data can be used to train and test intrusion detection models, thereby improving the generalization ability and detection accuracy of the models.

3. Proposed Method

This section first describes the benchmark datasets, then discusses the data preprocessing methods, and finally introduces the theory and model architecture of the proposed hybrid depth.

3.1 UNSW-NB15 dataset

UNSW-NB15 is a dataset created by Moustafa and Slay of the Australian Cyber Security Center (ACCS). It is an advanced dataset for evaluating the performance of Intrusion Detection Systems.

The dataset collects more than 100GB of real network traffic data by using the "IXIA PerfectStorm" automatic attack generation tool to attack multiple servers, covering nine different types of attacks, including 5% denial of service attacks. The dataset contains 2,540,044 instances, each with 49 columns of data, which are extracted and divided into five feature sets using Bro-IDS and Argus tools. Due to the imbalance of categories in the dataset, there are far more instances of normal traffic than attack instances, which may lead to overfitting problems in deep learning. To address this problem, regularization techniques are used to reduce overfitting when using deep learning to represent network traffic, thereby improving the generalization ability and accuracy of IDS.

3.2 Data Preprocessing Methods

Data preprocessing transforms raw data into a standardized form suitable for model processing through transformation and normalization techniques to optimize the performance of the Intrusion Detection System. In addition, null values are removed during the normalization process. In order to normalize larger values and reduce the impact of these larger values, the minimum-maximum scaling method is used to place the values between 0 and 1 for normalization.

The normalized equation is shown in (1).

$$f_{i,j} = \frac{f_{i,j} - \min(f_{i,j})}{\max(f_{i,j}) - \min(f_{i,j})} \tag{1}$$

In equation (1), $f_{i,j}$ represents the eigenvalue in the i-th row and j-th column of the data set matrix.

3.3 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) is an artificial neural network that is an efficient feature extraction tool and plays a key role in network data analysis. Since convolution and pooling are two different operations in the hidden layers of deep CNN, these hidden layers are called convolutional layers and pooling layers. In the convolutional layer, each filter generates a feature vector by sliding over the entire input data.

The equation for convolution calculation in a convolution layer is shown in (2).

$$y_i(k) = \sum_{n=0}^{N-1} x_i(n) h(k-n) \tag{2}$$

Where N is x_i the number of elements in the input vector, h is the filter, and n is y_i the element at position k in the output vector.

CNN greatly improves processing speed and efficiency through its unique weight sharing feature. The weights of these filters are shared throughout the network, which means that CNN requires fewer training parameters than other neural networks. This weight sharing not only reduces the complexity of the model, but also speeds up the training and reasoning process, making CNN the model of choice for processing large-scale network data. By alternating between convolutional layers and pooling layers, CNN can effectively identify patterns in one-dimensional, two-dimensional, and even three-dimensional data . In the pooling layer, the pooling operation reduces the output dimension to reduce computational costs and avoid overfitting. Commonly used pooling operations are divided into two categories, namely maximum pooling operations and average pooling operations.

In one-dimensional data, the equation for the maximum pooling operation is shown in (3).

$$r_i = \max\{y_i(p) : y_i(p+q-1), p = 1, q, 2q, \text{L}\ , nq, 1 \le n, q \le N\} \tag{3}$$

In one-dimensional data, the equation for the average pooling operation is shown in (4).

$$r_i = average\{y_i(p) : y_i(p+q-1), p = 1, q, 2q, \text{L}\ , nq, 1 \le n, q \le N\} \tag{4}$$

Where q represents the size of the filter, p represents the starting index, nq represents the ending index, and r_i represents the output vector.

3.4 Improved LSTM

Long Short-Term Memory Network is another type of neural network with feedback connections, which is usually used to process images, videos or speech. The common LSTM network consists of

a memory unit and three regulators to control the information flow within the LSTM unit. The three regulators are three AND-OR gates, namely the input gate, output gate and forget gate.

The dropout technology is used to regularize the LSTM neural network model, randomly shield some neurons, hide the LSTM weight matrix, and randomly discard some weights during model training to prevent the model from overfitting.

The equation formula of the LSTM output gate is shown in (5).

$$y_t = \sigma(Wx_t + (M \times U)h_{t-1}) \tag{5}$$

Where M represents the binary matrix mask corresponding to the data features. Each element mask M is changed separately to achieve the training of the improved LSTM model.

3.5 Fusion Deep Learning Model

The deep CNN-LSTM model is a hybrid deep learning architecture that combines the advantages of deep CNNs and improved LSTMs for Intrusion Detection of real-time network data traffic. The fused Deep Learning model consists of two one-dimensional convolutional layers, a one-dimensional maximum pooling layer, an improved LSTM layer, and a fully connected layer.

The calculation equation of the Rectified Linear Unit (ReLU) activation function used in the convolutional layer is shown in (6).

$$\sigma(\chi) = \max(0, \chi) \tag{6}$$

The output of the Maximum-Pooling layer is passed to the improved LSTM layer. The improved LSTM layer learns to extract the dependencies between data and randomly discards some weights to avoid overfitting. The output of the LSTM layer is passed to the fully connected layer, which contains the SoftMax activation function, which is used to detect and classify network intrusion data.

The calculation equation of the SoftMax activation function is shown in (7).

$$p_{i=} \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_i}} \tag{7}$$

Where x represents the input, p is the output value, the p value is between 0 and 1, and the sum of the p values is equal to 1.

The key parameters of the model are the number of CNN filters, the number of iterations, the learning rate, the number of improved LSTM hidden units, the drop-connect ratio, the batch size, and the maximum pooling size. All of these parameters are optimized through trial and error during training.

3.6 Evaluation method

In order to verify the performance of the CNN-LSTM model in Network Intrusion Detection, the following indicators are used to evaluate the experimental results.

3.7 Metrics

Metrics are important tools for evaluating model performance. These metrics can be used to evaluate the performance of the model.

True Positive(TP). TP means the number of positive classes correctly predicted as positive classes.

True Negative(TN). TN means the number of negative classes correctly predicted as negative classes.

False Positive(FP). FP means the number of negative classes that are incorrectly predicted as positive classes.

False Negative(FN). FN means the number of positive classes that are incorrectly predicted as negative classes.

Accuracy rate. Accuracy rate refers to the ratio of correctly predicted samples to the total number of samples, which indicates the overall predictive ability of the model.

The calculation equation of accuracy rate is shown in (8).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

Precision. Precision is a measure of the ratio of correctly classified samples to the total number of TP and FP.

The calculation equation of precision is shown in (9).

$$precision = \frac{TP}{TP + FP} \tag{9}$$

Recall rate. Recall rate is used to measure the ability of the model to identify positive samples, indicating the proportion of correctly identified samples in all actual positive samples.

The calculation equation of recall rate is shown in (10).

$$recall = \frac{TP}{TP + FN} \tag{10}$$

F1 score. F1 score is the harmonic mean of precision and recall and is used to measure the overall performance of the model.

The calculation equation of F1 score is shown in (11).

$$F1 - score = 2 \times \frac{precison \times recall}{precison + recall} \tag{11}$$

3.8 Experimental Results

The holdout testing technique is used to evaluate the model, where the UNSW-NB15 dataset is divided into a training set and a test set, where 70% of the data is used for training and 30% of the data is used for testing.

During training, the initial range of model hyperparameters was selected, and then the model hyperparameters were adjusted by trial and error. Finally, the optimal configuration was determined to be 50 epochs and a learning rate of 0.005 hyperparameter configuration.

The model converges stably during the training process and effectively suppresses overfitting, thereby improving the accuracy of network intrusion detection and reducing the loss of network intrusion, as shown in Figure 1.

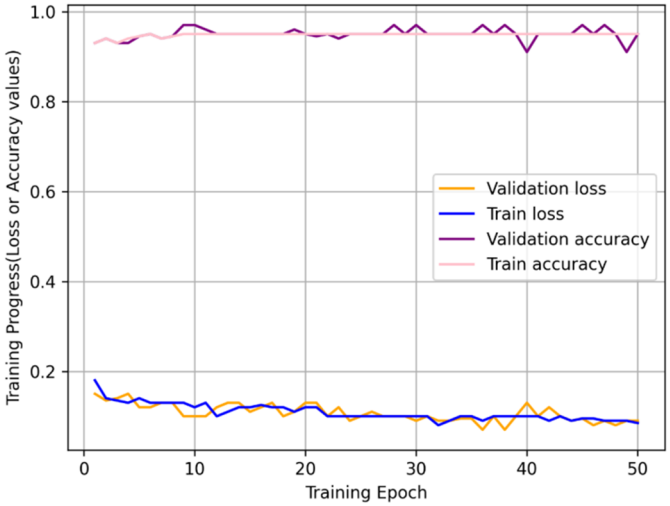

Figure 1Loss rate and accuracy during model training

The confusion matrix verifies that the model has excellent classification performance and detection quality on the UNSW-NB15 test set, with an accuracy of 95.16% and 96.03% and an F1 score of 0.97 respectively.

The performance indicators for normal and abnormal classification are shown in Table 1, and the performance indicators for normal and other types of attack classification are shown in Table 2.

Table 1 Performance indicators of normal and abnormal classification

	precision	Recall	F1-score	Accuracy

Normal	0.98	0.99	0.98	95.16%
Abnormal	0.94	0.82	0.88	
Weighted avg	0.97	0.97	0.97	

Table 2 Performance indicators for normal and other types of attack classification

	Precision	Recall	F1-score	Accuracy
Normal	1	1	1	96.03%
Exploits	0.64	0.8	0.71	
DoS	0.32	0.27	0.29	
Backdoor	0.5	0.07	0.12	
Analysis	0.44	0.09	0.15	
Fuzzers	0.71	0.61	0.66	
Generic	1	0.99	0.99	
Reconnaissance	0.93	0.77	0.84	
Shellcode	0.82	0.79	0.81	
Worms	0.5	0.09	0.15	
Weighted Avg	0.98	0.98	0.98	

Compared with existing models, the proposed model has better accuracy on the UNSW-NB15 dataset and can identify new types of network intrusions. The experiments were run on standard hardware configurations, and Table 3 shows that the average execution time of the model is lower than that of existing deep learning models, making it suitable for real-time intrusion detection systems. The main assumption of this comparison is that the model can be trained in offline mode and can detect Network Intrusion attacks in online mode.

Table 3 Average execution time of model checking

Model	Average execution time
CNN model	0.00372
LSTM Model	0.00298
CNN-LSTM Model	0.002383

On the UNSW_NB15 training set, 10-fold cross validation is used to compare the accuracy of the model. Table 4 shows the accuracy of the model for normal network traffic detection and abnormal network traffic detection.

Table 4 Accuracy of normal and abnormal traffic detection of the model

Model	Correctly Classified Instances	Incorrectly Classified Instances	Accuracy(%)
CNN model	212,007	24,316	88.531
LSTM Model	223,786	19,214	91.414
CNN-LSTM Model	229,173	7,150	96.880

By analyzing the results in Tables 1 to 4, it is concluded that the error rate of the CNN-LSTM Model is lower , while the error rates of other models are higher . The overall performance of the CNN-LSTM Model is better than other existing models.

4. Conclusions and future work

In this paper, a new Deep Learning model is proposed, namely a fusion model of a deep convolutional neural network and an improved long short-term memory network (CNN-LSTM), for

Network Intrusion Detection in a big data environment. The model first uses deep CNN to extract key features in network traffic data, and its weight sharing mechanism significantly improves the efficiency of data processing. In the model training phase, dropout technology is used to avoid overfitting of the training data by randomly shielding some neurons. In order to further capture the complex dependencies between features and enhance the generalization ability of the model, an improved LSTM network is introduced to effectively alleviate the overfitting problem through a weighted dropout mechanism while maintaining the sensitivity of the model to time series data. The hyperparameters of the model are carefully adjusted and optimized through experiments to ensure that the model achieves the best performance in actual network intrusion detection applications. Experimental results on the public UNSW-NB15 data set show that the CNN-LSTM model achieved a high accuracy of 95.16%, fully proving the effectiveness and superiority of this model in network intrusion detection tasks. In the future, we plan to continue to optimize this model and apply it to more complex and larger-scale data sets for further testing and analysis, with a view to developing an efficient Network Intrusion Detection System that can run in real time and provide powerful technology for the field of network security.

Acknowledgements

This research is partially supported by 2022 China University Research Innovation Fund - New Generation Information Technology Innovation Project(Grant No. 2022IT148), 2023 Chongqing Educational Science "14th Five-Year Plan" Special Key Topics for Education Reform(Grant No. K23ZG3050063).

References

[1] Hore S , Ghadermazi J , Shah A ,et al.A sequential deep learning framework for a robust and resilient network intrusion detection system[J].Computers & Security, 2024.

[2] Zhang X , Pan Y , Wu T ,et al.Brain age prediction using interpretable multi-feature-based convolutional neural network in mild traumatic brain injury[J].NeuroImage, 2024.

[3] Myeong-Joon K , Hyun-Jik C , Chul-Goo K . LSTM-AE based condition monitoring for reciprocating air compressors considering on/off characteristics[J].Journal of Mechanical Science and Technology, 2023, 37(12):6287-6295.

[4] Kayyidavazhiyil A . Intrusion detection using enhanced genetic sine swarm algorithm based deep meta-heuristic ANN classifier on UNSW-NB15 and NSL-KDD dataset[J].Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology, 2023, 45(6):10243-10265.

[5] Hssayni E H , Joudar N E , Ettaouil M .An adaptive Drop method for deep neural networks regularization: Estimation of DropConnect hyperparameter using generalization gap[J].Knowledge-based systems, 2022.

[6] Sohn I. Deep belief network based intrusion detection techniques: A survey[J]. Expert Systems with Applications, 2021, 167: 114170.

[7] Zhang X , Jiang L , Yang D ,et al.Urine Sediment Recognition Method Based on Multi-View Deep Residual Learning in Microscopic Image (vol 43, 325, 2019)[J].Journal of medical systems, 2020(4):44.

[8] Saxena D, Cao J. Generative adversarial networks (GANs) challenges, solutions, and future directions[J]. ACM Computing Surveys (CSUR), 2021, 54(3): 1-42.