Network Anomaly Traffic Detection Method Based on Multi-SAE and LSTM

Shuai Guo, Bowei Chen and Yang Su*

College of Cryptographic Engineering, Engineering University of PAP, Xi’an, Shanxi, China
Email: 1028618940@qq.com; 867106629@qq.com; *suyang75@126.com

Abstract. For the purpose of faster and more accurate anomalous traffic detection with increasing classes of data traffic in the network, this paper proposes a new anomalous traffic detection method based on stacked auto-encoders and a long short-term memory network model. The method uses Multi-SAE to extract the effective features of sequential traffic, which is obtained by concatenating multiple stacked auto-encoders, and a long short-term memory network to extract the temporal structure of the effective features, with the Multi-SAE and the long short-term memory network in a back-and-forth tandem structure. To further improve the efficiency of the detection, redundant MAC addresses are also removed in the pre-processing. From the experimental results, this paper achieves effective detection of twenty types of data traffic with an accuracy rate of 98.25%, which is higher than that of the same category of research by nearly 2 percentage points, and the parameters of precision, recall and f1-score also reach over 96%, improving the detection results.

Keywords. Deep learning; data stream; stacked auto-encoder; long short-term memory network; anomalous traffic detection.

1. Introduction

In recent years, the rapid development of network technologies represented by the Internet of Things, cloud computing, blockchain, 5G, etc. has led to a dramatic change in the network environment. Various applications are developing rapidly and the amount of traffic in the network is increasing, both in terms of the type, amounts and privacy of data traffic. These technologies have dramatically improved the development of society. But at the same time, it also brings new problems. The huge amount of network traffic and the extremely high transmission rate make it difficult to monitor the network effectively, and malicious traffic such as spam, Trojan, phishing websites and password theft frequently appear, causing great harm to society.

Previously, port-based and signature-based methods were used to address this issue. However, these two methods are no longer useful due to technological developments. New machine learning methods have emerged that can be more widely used, but also show a lack of performance in the face of complex network environments, for which researchers have proposed different models and methods. The commonly used methods are classification-based methods, which are subdivided into one-class classification methods and multi-class classification methods. One-class classification is used to identify target samples, and anything that does not belong to the target sample is marked as a negative sample. Multi-class classification methods are used to classify all categories of data as samples under imbalance. In this paper, we investigate multi-class classification methods. In terms of model selection, this paper improves on the temporal and spatial features in currently research, and
proposes the use of multiple Stacked Auto-Encoders [1] combined with LSTM to build models. The paper main work is:

1) Using the feature that SAE can retain the effective features to reduce the loss value, multi-SAE is used to extract sample features of different representations. Then Multi-SAE is combined with LSTM model to construct an anomalous traffic detection model with a combination of spatio-temporal features. Twenty classifications of network traffic are achieved, with accuracy, precision, recall and f1-score parameters reaching over 96%.

2) In order to compare the effects of different models, a set of comparison experiments was constructed, one model is Multi-SAE and LSTM, which was proposed in this paper, and another model is CNN and LSTM, which was obtained from other. The simulation experiments demonstrate that the combined Multi-SAE and LSTM model has a higher accuracy parameter than the CNN and LSTM models about two percentage points, while f1-score parameter is almost same.

3) In this paper, the redundant features of the data stream are removed from the training, and the MAC address is found to be a redundant feature through the packet transmission rule, which further improves the efficiency of the training.

2. Related Work

Wang [2] combined neural networks and deep learning to construct an unsupervised learning SAE method, which performs well in feature learning, protocol classification, anomaly protocol detection and unknown protocol identification. In the protocol classification of the 25 most widely used protocols, its precision and recall are above 95%, which can be fully applied in practice. In feature learning, the dimensionality of the data can be compressed to extract key features. After experiments, the top 25 features and the top 100 features in the data stream that have the greatest impact on classification are basically clustered in the first 200 bytes of the payload, and the 300 features that are least useful for classification are basically clustered in the middle and rear of the payload.

Wang et al. [3] conducted experiments using end-to-end convolutional neural networks and compared the experimental results of one-dimensional convolutional neural networks and two-dimensional convolutional neural networks. It was concluded that 1D-CNN (One-dimensional convolutional neural network) has better results for sequence-type data. The data is pre-processed by treating the bytes as pixels and processing the packets in the same way as pictures. The specific operation is: firstly, the data set is divided into data streams using five tuples (source IP, destination IP, source port, destination port, protocol), then the first 784 bytes of each data stream are intercepted to generate pictures, and the lack of the part is complemented by 0. Finally, the generated pictures are input into the deep neural network for learning. There is no explanation for the direct use of 784 bytes in the selection of feature length in the literature, but this paper believes that this value is too large and can be reduced to some extent.

Rezaei et al. [4] attempted to use the first six packets of a data stream, and selected the first 256 bytes of each packet as features for 80 classification by using a one-dimensional convolutional neural network, eventually achieving an accuracy rate of 84% to 98%. However, the paper removes some of the confusing fuzzy traffic, which is generated by web application modules, but it is different from normal network transmission data, such as the advertising module that comes with Google. This paper argues that this process removes a lot of the confusing traffic and improves its accuracy to some extent. Although CNN and LSTM were used in the second experiment to detect the ambiguous traffic, there were only seven types of them. In this paper, we believe that there is much space for improvement.

Reference [5] proposes an end-to-end DFR framework that automatically learns from the raw traffic and contains three components: CNN, LSTM and SAE. The three components are structured in parallel, with each network processing the dataset individually and then combining their processed features to synthesize the traffic detection. The advantage of this DFR model is that it contains different sub-networks, allowing different features to be extracted. This is a significant improvement over the previous model which used only one model to extract features, but it also has the
disadvantage that three networks are deep learning model structure, which inevitably requires the use of a larger dataset to achieve better performance, and the combination of the two will inevitably result in slower processing efficiency during training, which is a strong constraint on the method.

Zhang et al. [6] proposed a model combining multi-scale CNN and LSTM. The model uses multiple convolutional kernels, each with different sizes, when collecting spatial features from a dataset, and combining them can yield multiple sets of local features for more accurate recognition. After processing with CNN, the temporal features are also processed with LSTM, and finally the spatial and temporal features are used in combination for classification. This processing effectively reduces the false alarm rate of the detection system and improves the overall performance. In addition to this, reference [7] also uses the framework of CNN and LSTM to extract temporal and spatial features for the detection of abnormal traffic. The CNN and LSTM model is very widely used in current applications because it has a great advantage in extracting diversity features of samples, and this paper is an attempt to further improve it while retaining that advantage.

Reference [8] reduced the length of the data stream from 784 bytes to 529 bytes and then used a convolutional neural network model to classify the 784-byte and 529-byte datasets separately. The same classification results were obtained, demonstrating that both have the same effect, that is to say, the features removed from 784 bytes to 529 bytes are redundant features. In addition to this the article gives an explanation for the use of 529 bytes when performing feature selection. The paper goes further by removing the MAC addresses contained in the data traffic and setting all the deleted bytes to 0 in order to facilitate data processing, leaving the total length at 784 bytes.

From the above studies, the extraction of features started with a single model, then changed to a combination of multiple models in parallel, and now to a combination of CNN and LSTM in tandem. Feature extraction has transitioned from a single spatial feature to a combination of temporal and spatial features, with the aim of extracting more accurately sample features, but there is still space to improve in this method in the following directions. Firstly, previous research has focused more on the framework of temporal features and spatial features, but spatial feature is a type of feature, so the comprehensive extraction of effective features is the most important factor in recognition detection, so this paper uses the function that SAE can encode and decode the data to screen the effective features of the data, while keeping the loss rate minimal and retaining the effective features as much as possible. Secondly, the use of multiple SAEs for feature extraction, which are different and the features extracted are diverse, is very important for traffic detection. Thirdly, in terms of the feature screening, previous studies have also attempted to reduce the feature dimension. This paper builds on previous research to argue that MAC addresses do not work for classification, and that MAC addresses are censored to further reduce redundant bytes.

3. SAE and LSTM Based Detection Models
In this paper, we use a model combining multiple stacked auto-encoders and a long short-term memory network [9] to learn the network traffic dataset, figure 1 is the model structure. The data is first fed into a stacked auto-encoder network, which uses encoding and decoding to retain as many valid features as possible and reduce the loss of features, and then uses the compressed valid features to feed the results into a long short-term memory network, which can use time steps to stack states until all the time steps are cycled through to output the final results, and finally performs the detection of 20 classes of traffic in the Dense layer. Using this approach allows for a more accurate characteristic of the original input data.

Stacked auto-encoders contain similar functions to CNNs in that both can reduce the dimensionality of data features, but they differ in that CNNs use their pooling function to select the most locally significant features, often ignoring less important features, even if they are still important to classification, whereas stacked auto-encoders use their own coding and decoding function to make the input data equal to the output data. This approach retains as many valid features as possible and selects features from a global perspective, with less feature loss than a CNN. In addition, the use of multiple SAEs allows the extraction of different dimensions of the network features, which greatly
improves the detection capability of the model, and therefore the effective features retained in this approach are considered to be better than those of CNN.

![Detection model based on Multi-SAE and LSTM.](image)

**Figure 1.** Detection model based on Multi-SAE and LSTM.

Long short-term memory networks are a variant model of Recurrent Neural Network [10, 11], which can process sequence-type data by traversing all elements of the sequence using time steps and continuously overlaying content-related information to preserve memory, with the latter features appearing on top of the former ones. This temporal sequential relationship is used to better capture the sequence feature characteristics; the state of the recurrent neural network is reset between the processing of two different independent sequences. And compared to RNNs, long short-term memory networks go a step further by solving the gradient disappearance problem in RNNs and can carry a portion of the information to be transmitted to a later time step and used directly when needed. Compared to other studies that only use convolutional neural network models to obtain the spatial structure of the input data, this approach builds on it and captures the temporal structure of each effective feature more accurately, which is a significant enhancement for sample detection.

After building the model, the SAE needs to be trained firstly. The encoder and decoder parameters are trained using the input equals output property. At this point, the resulting parameters retain the maximum number of valid features from the dataset with the minimum loss value. The encoder part of the model is then intercept and the encoded features are obtained by feeding the original data to the encoder. These features are also arranged in a sequential order and can therefore be learned using the LSTM model. It also contains significantly less dimensionality than the original data, which improves the efficiency of the model processing. After inputting these effective features into the LSTM, the state is gradually superimposed using time steps, and finally recognition and detection is performed.

4. Model Parameter Selection Strategy

4.1. Strategies for Setting Parameters for Each Layer

The model used in this paper requires particular attention to the coding layers of the stacked autoencoders. The reason for this is twofold; firstly, multiple SAEs act together. These SAEs are different from each other, but they all have an input of 784 neurons and an output layer of 256 neurons. This is to maximise access to the feature representation. The number of neurons in the hidden layer between the input and output layers varies from SAE to SAE. The purpose of this is to obtain different feature representations during the compression process. Secondly, if the neurons in the input and output layers of the encoder are too different, their losses will also be larger, which is not conducive to the use of features. The aim of the settings is to both reduce useless features and to obtain more effective feature representations.
4.2. Packet Header Feature Selection Strategy
Packet headers contain redundant features such as MAC addresses, which need to be removed. The packet is transmitted through many of transit nodes such as switches and routers, where the packet information is read to determine the MAC address of the next hop based on the IP address, and then the packet is encapsulated, while the source and destination MACs are constantly changing with the next hop.

Therefore, the source MAC and destination MAC in the same network segment can correspond to the source IP and destination IP one by one, in this case adding MAC address can certainly play a role in the classification of network applications, this kind of representative data set is ISCX 2016. In this case, the MAC address cannot accurately represent the relationship between the source and destination addresses, and the two exist at the same time but cannot correspond, which will inevitably become an interfering term in deep learning, so it must be filtered. Removing the MAC part from the feature selection is feasible without affecting its accuracy and its feature dimensionality is reduced, which is an enhancement to the current study.

5. Simulation Experiments

5.1. Experimental Setup and Indicators
The dataset used for the experiments is USTC-TFC2016, which contains 10 types of normal traffic and 10 types of malicious traffic. The experiments models are two, one of the models is multiple SAE and LSTM, was proposed by this paper, and the other model is CNN and LSTM, was proposed from reference [7].

In the training and testing samples, the initial samples are divided into positive and negative samples, with the positive samples being the target samples and the negative samples being the non-target samples. The positive sample keeps changing as the detection target varies, especially in the case of multi-category samples. There is a possibility that a sample is incorrectly identified after detection, so there are four cases after detection, namely TP, FN, FP and TN. In order to assess the merit of the model, four parameters, accuracy, precision, recall and f1-score, are selected in this paper. Higher values of these four parameters indicate better recognition.

5.2. Space Considerations
As can be seen in figure 2, all parameters performed relatively well in this experiment, reaching 96% or more, which indicates that the model is more accurate in extracting effective features when performing sample detection. The results of Multi-SAE and LSTM models are 98.25% and CNN and LSTM models are 96.37%, a difference of about two percentage points between the two types of models, which is a large difference, indicating that Multi-SAE and LSTM models play a better role in detecting traffic and are more comprehensive in extracting features. The difference between the two types of models is about two percentage points.

In terms of precision, the SAE and LSTM models proposed in this paper are worse than the previously proposed CNN and LSTM models, with values of 96.58% and 97.44% respectively, a difference of about 1 percentage point, while the recall parameters have values of 99.2% and 97.11% respectively, a difference of 2 percentage points.

The F1-score can show the relationship between PRECISION and RECALL to prevent mistake due to overly prominent individual parameters. From the graph, we can see that both models are above 97%, indicating that both models perform better and have no major problems, with values of 97.87% and 97.14% respectively, with a difference of about 0.7 percentage points between the two, which is a small gap that can be regarded as a normal error fluctuation, i.e., it indicates that the two models do not differ significantly in their performance on precision and recall.

In summary, the Multi-SAE with LSTM model is about two percentage points higher in accuracy and 0.7 percentage points higher in f1-score parameter than the CNN with LSTM model. It proves that the proposed model in this paper has better performance in traffic detection.
6. Simulation Experiments

In this paper, a new method for detecting abnormal traffic is proposed to solve the traffic security problem in the current complex network environment. The method makes use of the feature that SAE can extract effective features comprehensively, builds Multi-SAE network to extract features of different dimensions effectively, and combines LSTM to extract temporal features to build a model that can implement detection for twenty types of data traffic. Compared with the previous model of CNN and LSTM, the accuracy is improved about 2 percentage points to 98.25%, and the precision and recall parameters also reach over 96%. In addition to this, the role of MAC addresses in the data stream for recognition detection is also discussed, reducing the feature bytes used and improving the processing efficiency. However, there are also some shortcomings in this paper, one is that the model is not streamlined enough, especially Multi-SAE because of the excessive use of SAEs makes the model more bulky and not efficient in processing data; the second is that the extraction of features by the SAE network is not accurate enough, and further improvement can be considered by combining the features of convolutional layers. These two aspects will be investigated in the next step.

References

[1] Mandavifar S and Ghorbani A A 2019 Application of deep learning to cybersecurity: A survey Neurocomputing 347 (JUN.28) 149-176.
[2] Wang Z 2015 The applications of deep learning on traffic identification BlackHat USA 24 (11) 1-10.
[3] Wang W, Zhu M, Wang J, et al. 2017 End-to-end encrypted traffic classification with one-dimensional convolution neural networks Proceedings of the 2017 IEEE International Conference on Intelligence and Security Informatics (ISI) pp 43-48.
[4] Rezaei S, Kroencke B and Liu X 2019 Large-scale mobile app identification using deep learning IEEE Access 8 348-362.
[5] Zeng Y, Gu H X, Wei W T, et al. 2019 Deep-full-range: A deep learning based network encrypted traffic classification and intrusion detection framework IEEE Access 7 45182-45190.
[6] Zhang J, Ling Y, Fu X, et al. 2020 Model of the intrusion detection system based on the integration of spatial-temporal features Computers & Security 89 101681.
[7] Kim T Y and Cho S B 2018 Web traffic anomaly detection using C-LSTM neural networks Expert Systems with Applications 106 66-76.
[8] Guo S and Su Y 2020 Encrypted traffic classification method based on data streams Journal of Computer Applications 41 (5) 1386-1391.
[9] Wang W, Sheng Y, Wang J, et al. 2017 HAST-IDS: Learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection IEEE Access 6 1792-1806.

Figure 2. Comparison of experimental results.
[10] Sherstinsky A 2020 Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network *Physica D: Nonlinear Phenomena* **404** 132306.

[11] Hochreiter S and Schmidhuber J 1997 Long short-term memory *Neural Computation* **9** (8) 1735-1780.