A New Intrusion Detection model based on GRU and Salient Feature Approach

Jian Hou, Fangai Liu and Xuqiang Zhuang

October 10, 2019

A New Intrusion Detection Model based on GRU and Salient Feature Approach

Jian Hou [1], Fangai Liu[1(✉)], Xuqiang Zhuang[1],

[1] Shandong Normal University, Shandong 250014, China
17051082@qq.com

Abstract. Gated Recurrent Unit (GRU) is a variant of a recurrent neural network, just like an LSTM network. Compared with RNN, the two networks have higher accuracy in processing sequence problems, and both of them have been proven to be effective in varieties of machine learning tasks such as natural language processing, text classification and speech recognition. In addition, the network unit structure of the GRU is simpler than the LSTM unit structure, which is more conducive to the training of the model. NSL-KDD datasets, which is the replacement of KDD cup 99, is still one of the datasets for measuring the effectiveness of intrusion detection models. In order to reduce the feature data dimension and combine the prior knowledge of computer network, a GRU intrusion detection method based on salient features (SF-GRU) is proposed. SF-GRU selects the distinctive features of response for different intrusion forms, and uses GRU network to identify the selected features to improve the efficiency of model detection. The experimental results show that compared with the traditional deep learning method, this proposal has higher accuracy and computational efficiency.
Keywords: Intrusion detection; Gate Recurrent Unit; Salient Feature selection; Prior Knowledge.

1 Introduction

Deploying network intrusion detection system (NIDS) in key nodes of the network is one of the important means to guarantee the security of cyberspace. At present, there are two kinds of commonly used network intrusion detection technologies, which are misuse detection technology based on prior knowledge and network anomaly detection technology based on network behavior. Among them, the former is mainly aimed at intrusion detection of known attack modes, and can not judge the network intrusion of location mode based on prior knowledge; anomaly detection technology based on network behavior is to distinguish normal and anomalous data by analyzing some characteristics of network flow, so as to realize the detection of intrusion behavior. Because of the inherent advantages of the latter technology, more and more scholars begin to study network intrusion detection from this perspective.

As one of the important technologies in the field of artificial intelligence, machine learning is also expanding it's application field. The research on network anomaly behavior detection combined with machine learning technology has also received

much attention. However, traditional machine learning is often inefficient in dealing with large-scale data. With the continuous development of Internet applications, network bandwidth is increasing, network transmission rate is increasing, and network application characteristics are increasing. Therefore, the efficiency requirements of network intrusion detection technology are also increasing. Compared with traditional machine learning technology, deep learning can handle higher-dimensional learning and more complex computing. [1].

In recent years, deep learning advantages in dealing with large-scale and high dimensional feature data is recognized by many scholars. Similarly, deep learning has also been applied to the study of cyberspace security [3]. Compared with traditional machine learning techniques, many scholars have studied the advantages of deep learning technology in the direction of network intrusion detection based on different algorithms. Among them, when performing network flow feature selection, DBN parameter debugging and pre-training can improve detection efficiency and reduce false positive rate [5]; In view of the training feature dimension, the paper [6] used the LSTM algorithm to select all features and adopt some features. The results show that the LSTM algorithm has certain advantages over other machine learning algorithms when using some feature training. Also as a variant of RNN, GRU networks improves network learning efficiency and detection accuracy to a certain extent compared to LSTM networks[11].

According to the time series characteristics of network data, this paper uses GRU network to simulate and train the NSL-KDD dataset and test the generated model. At the same time, in order to reduce the feature dimension under the premise of ensuring that the feature information is not lost, this paper proposes the GF-GRU (GRU based on Salient Features) algorithm, which is the GRU deep learning algorithm based on the feature selection. By selecting the salient features of the relevant intrusion behavior from the feature data, the input feature dimension is further reduced, and the complexity of the deep learning algorithm is reduced without affecting the algorithm evaluation index.

2 Related Work

The Gate Recurrent Unit (GRU) is an effective tool for processing sequence data, especially in the direction of sequence data learning of hidden features, GRU is easier to achieve reasonable classification. The GRU network is a kind of Recurrent Neural Network (RNN) [7]. In practical application, LSTM has a complex internal structure and a large number of parameters, which leads to the slow convergence of the training of cyclic neural networks. Cho et al. proposes a Gate Recurrent Unit (GRU), which has fewer model parameters and can transmit long-term information[8]. Compared with LSTM, GRU has fewer parameters, better performance and faster convergence speed.

LSTM solves the problem that the RNN often produces gradient disappearance when dealing with long-time training. The GRU network simplifies the LSTM memory unit and uses two gates(reset gate, update gate) to achieve selective memory of discrete time series long-term data. To facilitate understanding of the GRU algorithm, we use Figure 1 to describe the GRU forward computation unit process.

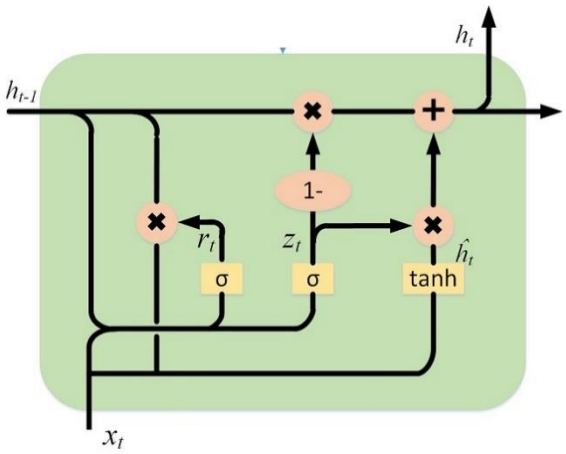

Fig. 1. GRU forward calculation unit

The equation of state for the two gates and the GRU memory cell in the figure is：

$$r_t = \sigma(W_r \cdot x_t + U_r \cdot h_{t-1} + b_r) \quad (1)$$

$$z_t = \sigma(W_z \cdot x_t + U_z \cdot h_{t-1} + b_z) \quad (2)$$

$$\hat{h}_t = tanh(W \cdot x_t + r_t * (U \cdot h_{t-1}) + b) \quad (3)$$

$$h_t = (1 - z_t) * \hat{h}_t + z_t * h_{t-1} \quad (4)$$

Where σ is the sigmoid function;r_t is the output of the reset gate, r_t controlling the effect of the output h_{t-1} of the hidden layer unit at the current moment on the time t；z_t is the output of the update gate, z_t used to determine the acceptance of the current input, similar to the input gate in LSTM, z_t enables the gradient to propagate effectively, effectively alleviating the gradient disappearance. The model is tested separately for different intrusion methods, and the final output is a two-category problem. Therefore, logistic regression is used for classification. The network error function selects the cross entropy loss function. In order to achieve the fast convergence of the gradient descent in the GRU network back propagation process, adaptive moment estimation (Adam) is adopted as the optimization algorithm.

3 Algorithm and Evaluation Index

The deep learning algorithm used in this paper is a recurrent neural network (RNN) with a gated loop unit (GRU). In order to reduce the relative complexity of the algorithm, this paper proposes different input features for different intrusion types.

3.1 Algorithm Model

It is well known that KDD datasets have redundant data. Even if NSL-KDD is greatly optimized with respect to KDD`99, this redundancy still exists, especially in the detection of a certain type of intrusion. According to the packet characteristics of computer networks, different types of intrusion packets have different network characteristics. For example, the four characteristics most relevant to DOS intrusion are "service", "flag", "src_bytes", and "count" [9]. Of course, testing with only these four features is not the most effective. Therefore, based on the prior knowledge of computer network and the existing research results, this paper selects different network features for different intrusion types, inputs GRU network to realize detection, and finally realizes the intrusion binary classification detection through the established recognition model.

The classification model proposed in this paper consists of four GRU network identification units (Fig. 2), each of which corresponds to the detection of an intrusion behavior. When the model is in the training phase, the training data is preprocessed and input into each GRU network identification unit, and each unit is individually trained according to the label. This process can achieve parallel computing, which means that the four subunits are trained separately at the same time, reducing the overall training time. According to the characteristics of the NSL_KDD dataset, in order to solve the data imbalance problem and improve the accuracy of model identification, the input data of each GUR network unit adopts different methods for preprocessing: the input data of the DOS attack recognition unit is all NSLs subjected to feature screening. KDD record; the input data of the PRIBE attack recognition unit is all data records after feature filtering and removing DOS attacks; the R2L attack recognition unit needs to be trained twice, first under-sampling the raw data with pre-processed negative label The R2L attack data accounts for about 40%. After the training is completed, the model is saved, then the whole data is used for migration learning, and the final model is trained. The input data of the U2R attack recognition unit is processed by the SMOTE algorithm on the preprocessed forward samples. Sampling, synthesizing new samples to alleviate class imbalance problems [12].

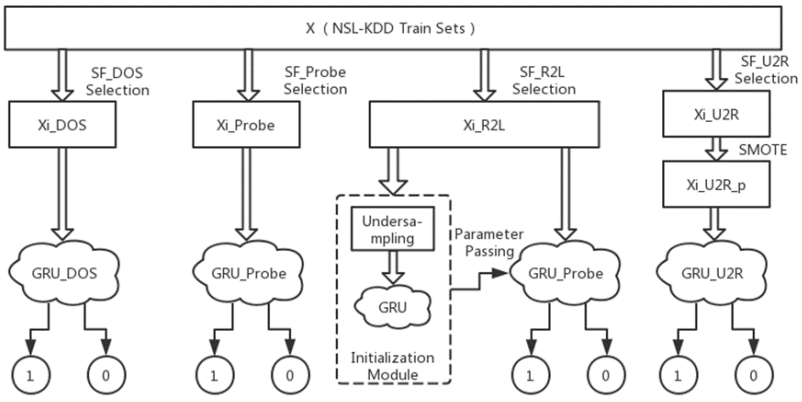

Fig. 2. SF-GRU Intrusion Detection Model

In the recognition stage, each data to be identified passes through four identification units in turn, and the classification of each data is recorded according to whether or not the intrusion data is used, and the binary classification detection of each intrusion behavior is realized. If the model is used to implement the binary classification detection of the entire intrusion behavior, the detection values of the four GRU network identification units are combined by means of "or" calculation. In general, any piece of data is detected as intrusion data in any unit, and the data is determined to be an intrusion.

3.2 Feature selection

From the observation and analysis of NSL-KDD data, each type of intrusion is reflected in the packet record, and the related feature elements are also different. Therefore, when analyzing a certain type of attack, only select this type of attack. A salient feature element that is reflected on the packet.

1. Denial of Service（DOS）

DOS attacks are common attacks that cause server crashes. Its common attack means is to send a large number of requests to the target server in a short time, and at the same time occupy a large amount of server resources, causing the server to fail to provide services. Therefore, the salient feature elements of the DOS attack must contain basic features such as service type, connection status, and target host unit time data volume. The existing research results show that the prominent feature elements belonging to the DOS attack are 11 feature elements such as 3-6, 8, 23, 29, 36, 38-40 in Table 1.

2. Probe

Probe attacks include IPsweep, nmap and so on, which belong to network scanning attacks or methods, so they have high correlation with network protocols, service types, and attack sources. Statistics show that the prominent feature elements of the Probe attack are 14 feature elements 2-6, 12, 29, 32-37, 40, etc. in Table 1.

3. Remote to Local（R2L）&User to root（U2R）

The amount of data for these two types of intrusion is relatively small, and the amount of information reflected in the feature elements is also scarce. According to the statistical analysis of the training dataset, the prominent feature elements used in the R2L intrusion are 14 feature elements such as 1, 3, 5, 6, 10, 24, 32, 33, 35-39, 41 in Table 1; U2R intrusion is adopted. The salient feature elements are 8 feature elements such as 3, 5, 6, 10, 14, 17, 32, and 33 in Table 1.

3.3 Feature data preprocessing

In the NSL-KDD dataset, there are three types of feature data, which are Boolean, symbolic, and continuous. Among them, the Boolean data and the percentage type

data in the contact data can be directly trained, such as the characteristics of 25-31 in Table 1.

For symbolic features, this paper uses ONE-HOT coding to map to multidimensional Boolean vectors. For example, in "protocol_type", tcp maps to [1,0,0], udp maps to [0,1,0], and icmp maps to [0, 0,1]. Similarly, "service" maps to a 70-dimensional Boolean vector and "flag" maps to an 11-dimensional Boolean vector.

For other continuity feature values, update with the following formula：

$$x_i = \frac{x_i - Min_{xi}}{Max_{xi} - Min_{xi}} \tag{5}$$

3.4 Evaluation Index

This paper mainly sets the evaluation index for the binary classification problem, and the evaluation visualization tool uses the confusion matrix, as shown in Table 1.

Table 1. Confusion matrix

Actual \ Predict	Positive	Negative
Positive	TP	FN
Negative	FP	TN

Where TP is the amount of data predicted to be intrusive; FP is the amount of data predicted to be intrusive but actually normal; FN is the amount of data predicted to be normal but actually intrusion data; TN is the amount of data correctly predicted to be normal.

According to the values of the four elements in the confusion matrix, this paper uses the following evaluation indicators:

— Accuracy(AC)

Accuracy indicates the percentage of records that can be correctly classified by the algorithm. This index is the most important indicator for evaluating the performance of the algorithm. The high accuracy is the most important embodiment of the algorithm. The calculation formula is:

$$AC = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

— Precision(P)

Predicting the correct intrusion record as a percentage of all predicted intrusion records is expressed as the accuracy of the algorithm. The calculation formula is:

$$P = \frac{TP}{TP + FP} \tag{7}$$

— Recall(R)

Predict the correct proportion of intrusions to all intrusions, expressed as the recall rate. The calculation formula is:

$$R = \frac{TP}{TP + FN} \tag{8}$$

Reducing the number of intrusion records detected may increase accuracy to a certain extent, but will reduce the recall rate. Therefore, it is necessary to consider both accuracy and recall rate in order to express the ability of the algorithm for intrusion detection. The joint calculation formula is:

$$F = 2PR/(P + R) \tag{9}$$

4 Experiment and Result Analysis

4.1 Dataset Selection

This paper uses the NSL-KDD dataset as the test dataset for the proposed algorithm. Compared to the original KDD`99 dataset, the NSL-KDD dataset has the following four advantages: First, the training set of the NSL-KDD dataset does not contain redundant records, so the classifier does not favor more frequent records; Second, there is no duplicate record in the test set of the NSL-KDD dataset, which makes the detection rate more accurate. Third, the classification rate of different machine learning methods changes within a wider range, which makes the accurate evaluation of different learning techniques more accurate. Effective; fourth, the number of records in training and testing is set reasonably, which makes the cost of running the experiment in the entire set of experiments lower. In addition, many researchers have done a lot of research on machine learning in the NSL-KDD dataset [8], so it is easy to obtain comparative data.

The NSL-KDD dataset contains the "KDDTrain+" training set of 125,973 data, the "KDDTest+" test set of 22,554 data, and the "KDDTest-21" test set of 11,850 highly difficult information. Each piece of data contains 41 features, 1 classification and 1 difficulty value. Three of the 41 features are non-numeric, they are "protocol_type", "service", and "flag", which need to be digitized during data preprocessing. Table 2 shows the 41 feature elements and their data types for each record in the NSL-KDD dataset.

Table 2. Features and types

Type of feature	Intrusion type
Numeric	(1)Duration,(5)Src_bytes,(6)Dst_btyes,(9)Urgent,(10)Hot,(18)Num_shells,(11)Num_failed_logins, (13)Num_compromised,(16)Num_root,(17)Num_file_creations,(19)Num_access_files,(20)Num_outbound_cmds, (23)Count,(24)Srv_count,(25)Serror_rate,(26)Srv_serror_rate,(28)Srv_rerror_rate,

	(29)Same_srv_rate,(30)Diff_srv_rate,(25)Rerror_rate,(31)Srv_diff_host_rate,(32)Dst_host_count, (33)Dst_host_srv_count,(34)Dst_host_same_srv_rate,(35)Dst_host_diff_srv_rate, (36)Dst_host_same_src_port_rate,(37)Dst_host_srv_diff_host_rate,(38)Dst_host_serror_rate, (39)Dst_host_srv_serror_rate,(40)Dst_host_rerror_rate,(41)Dst_host_srv_rerror_rate
Nominal	(2)Protocol_type,(3)Service,(4)Flag.
Binary	(7)Land,(8)Wrong_fragment,(12)Logged_in,(14)Root_shell,(15)Su_attempted,(21)Is_hot_login. (22)Is_guest_login

4.2 EXPERIMENTAL RESULTS

According to the above model and evaluation index, this paper carries out simulation experiments with NSL-KDD dataset. The main purpose of the experiment is to find the optimal hyperparameters of the GRU sub-network in the model, such as the optimal learning rate and the GRU hidden layer size. Then, determine the optimal hyperparametric comparison training set and test.

According to the research results in [10] and the similarity between GRU and LSTM network structure, the learning rate and hidden layer size of GRU network are independent of each other in the impact of the algorithm, that is, they can be debugged separately when adjusting the network. In this experiment, the learning rate is debugged separately for each GRU learning module. The results are as follows:

Table 3. Accuracy（AC）

type	Learning rate 0. 1%	Learning rate 0.05%	Learning rate 0. 01%	Learning rate 0.005%
DOS	0.9742	0.9806	0.9788	0.9729
Probe	0.9768	0.9848	0.9812	0.9755
U2R	0.9994	0.9995	0.9967	0.9993
R2L	0.9936	0.9984	0.9967	0.9979

Table 4. F value

type	Learning rate 0. 1%	Learning rate 0.05%	Learning rate 0.01%	Learning rate 0.005%
DOS	0.9638	0.9728	0.9702	0.9620
Probe	0.8693	0.9160	0.8967	0.8624
U2R	0.5627	0.5804	0.5891	0.5977
R2L	0.9011	0.9031	0.8967	0.8721

The second phase of the experiment is a fixed learning rate of 0.05%, which adjusts the size training model of the GRU hidden layer. The experimental results show that when the hidden layer size is 80, the model effect is optimal, and the specific experimental results are as follows (Table 5):

Table 5. fixed learning rate of 0. 05%

type	40 Hidden layer	60 Hidden layer	80 Hidden layer	100 Hidden layer
DOS	0.9529	0.9626	0.9728	0.9456
Probe	0.9119	0.9129	0.9364	0.9189
U2R	0.5804	0.5804	0.5717	0.5804
R2L	0.9026	0.9102	0.9186	0.9080

From the experimental results, the U2R type detection F value is low, the analysis from the original data is because the false positive rate is high, the high false positive rate is caused by the over-fitting of the model, but due to the data imbalance The accuracy of the model is still high.

Therefore, this paper chooses the learning model with the learning rate of 0.05% and the hidden layer of GRU as 80, and compares it with the accuracy of the traditional machine learning algorithm. The SF-GRU intrusion detection model is on the NSL-KDD dataset. The performance has certain advantages, the experimental results are as follows:

Table 6. Accuracy comparison of algorithms

type	SF-GRU	Random Forest	J48	SVM	CART
DOS	0.9812	0.9821	0.8248	0.9778	0.8894
Probe	0.9861	0.9762	0.8029	0.9074	0.8273
U2R	0.9995	0.9754	0.7394	0.9376	0.7308
R2L	0.9984	0.9681	0.8759	0.9182	0.8083

The comparison results show that the SF-GRU Intrusion Detection Model has an improvement in accuracy compared to the traditional machine learning algorithm. Meanwhile, due to the adoption of a simplified recurrent cell and the use of less dimensional input data in the model input, Sf-GRU intrusion detection model is superior in time complexity than traditional LSTM-based intrusion network detection model.

5 Conclusions

In the NSL-KDD dataset, each piece of data contains 41 features that are often redundant in the detection of certain aggressive behaviors. Therefore, by selecting the appropriate feature data to participate in the calculation through data preprocessing, not only can the detection rate not decrease with the decrease of the feature, but also the model training time can be effectively reduced.

Based on the prior knowledge of computer network and the existing research results, this paper applies GRU algorithm to the NSL-KDD dataset after feature selection, and proposes a GRU intrusion detection algorithm based on salient features (SF-GRU). Experiments show that this method has higher accuracy than traditional

machine learning methods, and compared with the original LSTM detection method, the model complexity is reduced and the algorithm efficiency is improved.

Acknowledgments

This work was supported by National Natural Science Foundation of China (61772321), CERNET Innovation Project (NGII20170508), and in part by Guangdong Province Key Research and Development Plan (Grant No. 2019B010137004), the National Key research and Development Plan (Grant No. 2018YFB1800701, No. 2018YFB0803504, and No. 2018YEB1004003), and the The National Natural Science Foundation of China (Grant No. U1636215 and 61572492).

References

1. Yin C, Zhu Y, Fei J, et al. A deep learning approach for intrusion detection using recurrent neural networks[J]. IEEE Access, 2017, 5: 21954-21961.
2. LeCun Y, Bengio Y, Hinton G. Deep learning[J]. nature, 2015, 521(7553): 436.
3. Yuan Z, Lu Y, Wang Z, et al. Droid-sec: deep learning in android malware detection[C]//ACM SIGCOMM Computer Communication Review. ACM, 2014, 44(4): 371-372.
4. Depren, ozgur, et al. An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks, Expert systems with Applications 29.4, pp.713-722,2005.
5. Gao N , Gao L , Gao Q , et al. An Intrusion Detection Model Based on Deep Belief Networks[C]//2014 Second International Conference on Advanced Cloud and Big Data (CBD). IEEE Computer Society, 2014.
6. Staudemeyer R C . Applying long short-term memory recurrent neural networks to intrusion detection[J]. South African Computer Journal, 2015.
7. Cho K , Van Merrienboer B , Gulcehre C , et al. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation[J]. Computer Science, 2014.
8. Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv preprint arXiv:1406.1078, 2014.
9. Staudemeyer R C, Omlin C W. Extracting salient features for network intrusion detection using machine learning methods[J]. South African computer journal, 2014, 52(1): 82-96.
10. Greff K , Srivastava R K , Koutník, Jan, et al. LSTM: A Search Space Odyssey[J]. IEEE Transactions on Neural Networks & Learning Systems, 2015, 28(10):2222-2232.
11. Agarap A F M . [ACM Press the 2018 10th International Conference - Macau, China (2018.02.26-2018.02.28)] Proceedings of the 2018 10th International Conference on Machine Learning and Computing, - ICMLC 2018 - A Neural Network Architecture Combining Gated Recurrent Unit (GRU) and Support Vector Machine (SVM) for Intrusion Detection in Network Traffic Data[J]. 2018:26-30.
12. Han H , Wang W Y , Mao B H . Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning[C]//Proceedings of the 2005 international conference on Advances in Intelligent Computing - Volume Part I. 2005.
13. Wang, Qianqian, et al. "Research on CTR Prediction Based on Deep Learning." IEEE Access 7 (2018): 12779-12789.

14. Z. Tian, W. Shi, Y. Wang, C. Zhu, X. Du, S. Su, Y. Sun and N. Guizani. Real Time Lateral Movement Detection based on Evidence Reasoning Network for Edge Computing Environment. IEEE Transactions on Industrial Informatics. 2019. Vol 15(7): 4285-4294.
15. Z. Tian, M. Li, M. Qiu, Y. Sun, S. Su. Block-DEF: A Secure Digital Evidence Framework using Blockchain, Information Sciences. 491(2019) 151-165. DOI: 10.1016/j.ins.2019.04.011.
16. Z. Tian, X. Gao, S. Su, J. Qiu, X. Du and M. Guizani. Evaluating Reputation Management Schemes of Internet of Vehicles based on Evolutionary Game Theory. IEEE Transactions on Vehicular Technology. 2019. Vol 68(6): 5971-5980.
17. Z. Tian, S. Su, W. Shi, X. Du, M. Guizani and X. Yu. A Data-driven Method for Future Internet Route Decision Modeling. Future Generation Computer Systems. 2019. Vol. 95, 212-220.
18. Q. Tan, Y. Gao, J. Shi, X. Wang, B. Fang and Z. Tian, Toward a Comprehensive Insight Into the Eclipse Attacks of Tor Hidden Services, IEEE Internet of Things Journal, vol. 6, no. 2, pp. 1584-1593, April 2019.