Learning Representations Using RNN Encoder-Decoder for Edge Security Control

Wei Guo 1, Hexiong Chen 1, Feilu Hang 1, Yingjun He 1, and Jun Zhang 2

1 Information Center, Yunnan Power Grid Co., Ltd, Kunming 650000, China
2 Department of Technical Development, Sichuan Wiscred Communication Co. Ltd, Chengdu 610043, China

Correspondence should be addressed to Jun Zhang; zhangjun@wiscred.com

Received 29 March 2022; Accepted 4 May 2022; Published 22 May 2022

Abstract

Whitelisting is a widely used method in the security field. However, due to the rapid development of the Internet, the traditional whitelisting method cannot promote the security of increasing Internet access. In recent years, with the success of machine learning in different areas, many researchers focus on the security of Internet access through machine learning methods. The most common form of machine learning is supervised learning. Supervised learning requires a large number of labeled samples, but it is difficult to obtain labeled samples in practical applications. This paper introduced an unsupervised deep learning algorithm based on seq2seq, which combined with the recurrent neural network and the autoencoder structure to realize an intelligent boundary security control mechanism. The main methods proposed in this paper are divided into two parts: data processing and modeling. In the phase of data processing, the access text table was coded with dicts, and all sequences were padded to the maximum. In the modeling phase, the network was optimized according to the principle of minimizing the reconstruction error. From the comparative experiments, the proposed method’s AUC on the public data set reached 0.99, and its performance is better than several classical supervised learning algorithms, proving that the proposed method has an efficient defense against abnormal network access.

1. Introduction

Since Maxwell condensed the laws behind the electromagnetic field with equations, information technology has ushered in rapid development. With the development of information technology, computer network has gradually formed the existing hierarchical structure. As a communication protocol, TCP/IP realizes the interconnection between various types of networks. However, the convenience brought by network interconnection also brings new problems and challenges to computer network security. Computer network security mechanisms can be roughly divided into encryption mechanisms, control mechanisms, and supervision mechanisms. This paper mainly discusses the security control field of network edge. The goal of traditional network edge security control technology is to control the access that only allows trusted traffic sources. The core of network security is the trust mechanism. The traditional model can prevent some network attacks, but the trust control of network attacks is not complete.

The edge security control mechanism proposed in this paper, which combines the recurrent neural networks and intelligent whitelist verification, is no longer limited to setting fixed and rule-based strategies but uses deep learning methods to automatically extract security features from network access and dynamically monitor the security status of each network request to deal with the increasingly complex network security situation intelligently.

The contributions of this paper are as follows:

(a) The core of security is the trust mechanism, but the traditional rule-based method is difficult to solve the existing trust problem. The edge security control mechanism of intelligent whitelist verification proposed in this paper can identify network access more dynamically than the traditional rule-based method to ensure network security.

(b) The edge security control mechanism method combined with recurrent neural network and intelligent whitelist verification proposed in this paper...
is an unsupervised learning method, which can dynamically mine potential attacks without labeled data. It works better in the case of a rapid increase in network flow and the one having difficulty obtaining artificially marked network access to abnormal data.

2. Related Works

Network edge security control needs to focus on data in three areas: user behavior, network flow behavior, and data packet characteristics.

(a) User behavior: In the field of network edge security control, user behavior refers to the statistics and analysis of accessing users, which are identified by the IP address of the device used, the certificate obtained from the network access authentication, etc., to find the laws of users accessing the network, server, or other information technology equipment, and combine these laws with the network edge security control to discover potential security risks in the current activities of users accessing the network and crossing the micro boundary of the network segment and provide a basic basis for the application of security control strategies.

(b) Network flow behavior: Network flow behavior is described by the multiparty characteristics, including the flow connection characteristics of the communication network, the evolution of the flow connection characteristics, the time-varying characteristics, and the evolution characteristics of the flow behavior.

(c) Data packet characteristics: The network connection characteristic information of the data packet includes the protocol (TCP, UDP, etc.) that the data packet belongs to, the source address, the destination address, the port number of the destination device (request type), the transmission direction of the data packet, and the data packet classification, etc.

The main types of network attacks are as follows:

(a) Denial of service attack: The target of this attack is to make the normal service computer system unable to provide functions normally due to malicious blocking.

(b) Network sniffing attack: The target of this attack is to collect information about the target network or port. Although this attack method will not cause the computer system to crash, it is often a preparation for subsequent attacks.

(c) U2R (User-to-Root): This unauthorized attack is mainly to gain super user privileges after sneaking into the system;

(d) R2L (Remote-to-Login): This attack is mainly to remotely control the computer permissions of others [1].

In the field of network edge security control, intrusion detection systems and firewall systems are traditionally used in combination. Intrusion detection mainly detects whether there is malicious access in the network, tampering with information in the system, or even the behavior causing the system to crash. The firewall system mainly analyzes the data packets of each data layer in the seven-layer model of the computer network and performs the corresponding operations according to the security policy of the firewall [2].

In the field of intelligent network edge security control, the methods can be divided into traditional rule-based methods and machine learning model-based methods.

2.1. Traditional Methods

Traditional methods mainly include packet filtering and application agents [3]. Packet filtering is a precipitation of knowledge formed by domain experts. However, the rules formed by domain knowledge have a defensive range delineated by the rules, and they cannot defend against all attacks [4]. In the case of a large amount of network flow, the technology of the application agent cannot achieve a good balance among normal system functions, service, and security detection. There is also network detection based on signal processing technology, which mainly uses a general likelihood ratio to detect signal anomalies [5]. However, this method also relies on manual experience.

2.2. Machine Learning Methods

Machine learning methods can be divided into supervised learning methods and unsupervised learning methods [6].

2.2.1. Supervised Learning Method

(a) Support vector machine (SVM): An SVM is an approximate model to minimize structured risk. Its core idea is to maximize the classification interval, and it is a supervised learning method combined with a hinge loss function [7].

(b) Robust SVM: Robust SVM adds a sparsity constraint on the basis of SVM [8]. For example, add the constraint of $L_1$ to the target solution: the sparsity constraint makes many elements of the target solution close to zero. Such characteristics will not only bring convenience to the solution and storage but also ensure the stability of the solution under the interference of input noise;

(c) Bayesian network: Bayesian network is a directed probability graph model for modeling uncertainty. It combines experts’ knowledge of state transition probability in the security field to detect network attacks [9];

(d) Decision tree: The decision tree model is based on a recursive method, using a certain splitting criterion to classify data, such as the information gain [10];

(e) Neural network: Neural network has a universal approximation to the function; that is, for a given objective function, there is a neural network structure and the corresponding activation function to approximate the objective function [11]. In the
2.2.2. Unsupervised Learning Method

(a) Mathematical statistics: This kind of method, such as the Chi-square test, Gaussian 3δ test, etc., mainly evaluates the abnormality of the sample through a probability distribution [17]. However, if the sampling method does not meet the Gaussian distribution or the presupposed distribution, it will affect the effect of the method;

(b) Principal component analysis: The principal component analysis method mainly uses the singular value decomposition in the matrix decomposition, and the singular value of the nonsquare matrix is used as the criterion for judging abnormalities [18]. In practice, the two components with the largest singular values are mainly used, that is, the singular values of the main principal component and the secondary principal component. If the ratio of the two singular values exceeds a certain threshold, the visit is judged to be abnormal. The main advantage of this method is that there is no need to make any statistical assumptions about the distribution, and the calculation efficiency is high. However, since there is no specific standard for the selection of scores, the application scenarios of the algorithm are subject to certain restrictions;

(c) Information theory: This method mainly uses the concepts of entropy, cross-entropy, and information gain in information theory to evaluate whether the model is suitable for new data sets [19];

(d) Hybrid model: This method mainly uses the general approximation of the mixed model to the distribution to fit the distribution [20]. Before the rise of deep learning methods, this type of method was widely used for anomaly detection. For example, if the Gaussian mixture model is approximated to the Laplace distribution, an infinite number of Gaussian components are required, which is limited in practical applications.

The effect of supervised learning methods is generally better than that of unsupervised learning, but supervised learning requires a large amount of labeled data, while unsupervised learning does not require a large number of labeled samples. Therefore, this paper proposed a control mechanism based on a recurrent neural network, combined with the unsupervised method of an autoencoder, to realize an edge security control mechanism based on intelligent whitelist verification.

3. Unsupervised Deep Learning Algorithm for Edge Security Control

The edge security control mechanism, which combines a recurrent neural network and intelligent whitelist verification, builds a deep neural network directly based on the request text of network access so that the deep neural network can automatically mine abnormal features and give the classification results of network access. This paper used a GRU-based recurrent neural network. This neural network has a self-encoder structure and uses an unsupervised method that does not require label data to achieve an intelligent whitelist algorithm with better results. The parameters of the GRU-based recurrent neural network are fewer than those of LSTM. As shown in the experiments, the GRU-based recurrent neural network has achieved a robust result. The process of an edge security control mechanism based on intelligent whitelist verification is shown in Figure 1.

3.1. Data Preparation and Preprocessing. The data accessed from the network are stored as text. The specific text format is shown in the experimental part of Section 4. The first is to encode the text into the corresponding number to facilitate subsequent program processing. There are some special symbols, which are mainly used to complete the sequences that do not meet the requirements of the longest text. Usually, the code is 0. They are used at the end and the beginning of the decoder to tell the decoder the start and end of the sentence and to represent words that do not appear in the dictionary, generally low-frequency words. Others are mapped to characters and numbers according to the given text table. The coding example of network access data is shown in Figure 2. The data cleaning part mainly uses regular expressions to extract network access topics [21].

3.2. Model. The network used in this paper is Seq2Seq based on gated recurrent units (GRU) combined with an autoencoder network architecture. The goal of this model is to learn the objective function $f: R^T \rightarrow R^T$, where $T$ represents the maximum length in the network access sequence. Seq2Seq is a method of encoder-decoder-based machine translation and language processing that maps an input sequence to output sequence with a tag and attention value. The idea is to use 2 RNNs that will work together with a special token and try to predict the next state sequence from the previous sequence. The whole process is shown in Figure 3.
The text data are time-series data as each character in the network access sequence is input to the GRU for processing. Therefore, the selected network structure is a classical cyclic neural network. Since the simple recurrent neural network is prone to gradient disappearance after multiple passes, there will be some problems with the model, which cannot be trained. This paper used GRU to process text to overcome the short-time dependence of a simple cyclic neural network. The GRU is like a long short-term memory (LSTM) with a forget gate, but has fewer parameters than LSTM, as it lacks an output gate. The illustration of GRU is shown in Figure 4, as there are reset gate and update gate [22]. The first $T$ gated recurrent units make up the encoder, and the encoding vector is obtained after the text passes through the encoder. The encoding vector is input into the decoder to obtain the probability value of each decoded text, and select the element with the largest probability value as the output. The structure of the decompressor is symmetrical to the encoder.

3.2.1. Calculation of GRU

(a) Candidate hidden state $\overline{h}_t$ is computed using the following equation:

(b) Hidden state $h_t$ is computed using the following equation:
\[ h_t = Z_t \ast h_{t-1} + (1 - Z_t) \ast \tilde{h}_t, \] (2)

where \( Z_t = \sigma(W_{xz}X_t) \) is the update gate and \( \sigma \) is the activation function.

In this paper, \( \sigma \) is the sigmoid function, and is defined as follows:
\[
\sigma = \frac{1}{1 + e^{-x}}.
\] (3)

3.2.2. Calculation of seq2seq. In Figure 3, the first \( T \) gated recurrent units constitute the encoder, and the last \( T \) gated recurrent units constitute the decoder. The whole structure can be summarized as follows:
\[
\begin{align*}
    h_t &= f_e(h_{t-1}, x_t), t \in [1, T], \\
    h_{T:t} &= f_d(h_{T:t-1}, \tilde{y}_{t-1}), t \in [1, T], \\
    \tilde{y}_{t-1} &= g(h_{T:t})t \in [1, T].
\end{align*}
\] (4)

\[ h_T = f_d(h_{T-1}, x_{T-1}) \] is the encoding vector, as the input to the decoder, \( f_e \) represents the encoder network composed of gate recurrent units, \( f_d \) represents the decoder network composed of recurrent units, \( g \) represents the last layer of the softmax classification network, and \( \tilde{y} \) represents the prediction output with the largest probability value.

The most commonly used optimization algorithm for neural network training is gradient descent. This paper used the unsupervised training method, so the training sample accessed by the network is \((x, y)\), where \( x \) is the network access text with sequence length \( T \), \( x = (x_1, \ldots, x_T) \), and \( y = x \) since the unsupervised training method adopted.

The length of the sequence data accessed by the network is generally inconsistent, and the sample space to be searched is relatively large. With the multiplication chain of probability theory in the following equation:
\[
p(x_{1:T}) = p(x_1)p(x_2|x_1)p(x_3|x_{1:2}) \cdots p(x_T|x_{1:T-1}).
\] (5)

With the loss function, the conditional probability at sequence \( t \):
\[
p_\theta(y_{1:T}|x_{1:T}) = \prod_{t=1}^{T} p_\theta(y_t|y_{1:(t-1)}, x_{1:T}).
\] (6)

Under the framework of unsupervised learning, the maximum likelihood estimation is used to train the model parameters \( \theta \), where \( \theta \) contains the weight \( W_{sh}, W_{hh}, b_h \).
\[
\tilde{\theta} = \argmax_\theta \sum_{n=1}^{N} \log p_\theta(y_{1:T}|x_{1:T}).
\] (7)

To solve the above optimal solution, the gradient descent method is generally used, therefore:
\[
\theta \leftarrow \theta - \frac{\partial}{\partial \theta} \sum_{n=1}^{N} \log p_\theta(y_{1:T}|x_{1:T}).
\] (8)

After training, the model can generate the most likely target sequence based on the input sequence:
\[
\tilde{y} = \arg\max_y p_\theta(y|x).
\] (9)

The above optimal solution can be obtained by beam search, and the beam search process is shown in Figure 5.

Firstly, generate \( m \), the most probable word codes, where the size of the beam search is \( m \). Then select the most probable sequence from the words with the size of the word code table.

After data preprocessing, the seq2seq network is constructed and trained, and the final result \( \tilde{y} \) is obtained. Use the difference \( \| \tilde{y} - x \| \) between \( \tilde{y} \) and \( x \) to judge whether it is normal network access. In Section 4, this paper will verify the effect of the algorithm through experiments.

4. Experiment

To verify the effectiveness of the edge security control mechanism proposed by this paper, we used a public data set for verification.

This paper presents an unsupervised intelligent whitelist verification method without label data by combining the encoder and decoder network structure with a GRU-based deep neural network. The evaluation index is to use the ROC curve to calculate the area under the ROC curve (AUC).

4.1. Experimental Configuration and Data Preparation. This experiment ran on a 4-core Intel(R) i7-4720HQ-cup @2601Mhz laptop with 16 GB of RAM. The public data set used is a bank’s network access dataset, of which 20,000 are normal access records and 1,000 are offensive access records. An example of the bank network access data are shown in Figure 6.
An example of a bank’s network access is as follows:

```
ST@RT
Thu, 15 Mar 2018 14:45:52 INFO
GET/vulnbank/index.html HTTP/1.1
Host: 10.0.212.25
Connection: keep-alive
Cache-Control: max-age = 0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.186 Safari/537.36
Accept: text/html, application/xhtml+xml,application/xml; q 0.9, image/webp, image/apng/; q 0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-US, en; q = 0.9
END
```

The deep neural network is trained on 20,000 normal access records and 1,000 access records with attacks. There are two hidden layers. The probability of dropout is 0.7, and the encoding layer output by the encoder is 64.

4.2. Experimental Results. The seq2seq model used in this paper essentially solves the binary classification problem. The ROC curve is the commonly used index for evaluating the binary classification problem. Therefore, this paper used the ROC curve to evaluate the experimental results. There are four outcomes of binary classifiers as follows:

1. True positive (TP): the classifier judges it to be an abnormal access, which is actually an abnormal access.
2. False positive (FP): the classifier judges it to be an abnormal access, which is actually a normal access.
3. True negative (TN): the classifier judges it to be normal access, which is actually a normal access.
4. False negative (FN): the classifier judges it to be normal access, which is actually an abnormal access.

Given a rejection threshold $\eta$, the ratio TPR as recall can be calculated, and the ratio FPR that was tested as false by mistake can be calculated.

$$\frac{FP}{FP + TN}, \quad \frac{TP}{TP + FN}, \quad \frac{TP + TN}{TP + TN + FP + FN}, \quad \frac{TP}{TP + FP}, \quad \frac{TP}{TP + FN}, \quad \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

The threshold $\eta$ can be determined by the purpose of the application.

In this paper, the unsupervised method based on seq2seq is experimentally verified on the bank data set. There are two hidden layers, the probability of dropout is 0.7, and the encoding layer output by the encoder is 64. Moreover, the proposed method is compared with several supervised learning algorithms, such as Support Vector Machine, XGBoost, GBDT, and Decision Tree. The experimental results are shown in Table 1. From Table 1, precision, recall,
Table 1: Experimental results.

| Algorithm   | Precision       | Recall | F1_score       | Accuracy_score |
|-------------|-----------------|--------|----------------|----------------|
| seq2seq     | 0.9995339       | 1      | 0.999771898    | 0.999556221    |
| Decision Tree | 0.998629824   | 0.997718458 | 0.998173933 | 0.996535297  |
| SVM         | 0.999543692     | 0.999543692 | 0.999543692 | 0.999133824  |
| GBDT        | 0.999088007     | 0.999771846 | 0.99942981   | 0.99891728   |
| XGBoost     | 0.9995439       | 1      | 0.999771898    | 0.999566912   |

$F_1$ score, and accuracy score of the unsupervised method based on seq2seq are 0.9995339, 1, 0.999771898, and 0.999566221, respectively. This result shows that this method is better than Decision Tree, support vector machine, and GBDT. Specifically, its precision, $F_1$-score, and accuracy are 0.1%, 0.15%, and 0.3% higher than Decision Tree’s, respectively.

In the bank data set, the ROC curve and AUC of the edge security model proposed by this paper are shown in Figure 7. The AUC of the unsupervised method based on seq2seq is 0.996. Moreover, its AUC is better than the one of Decision Tree and GBDT, since its AUC is 0.01 higher than Decision Tree’s, and 0.005 higher than GBDT’s. From the above comparative experiments, the method proposed by paper can identify abnormal network attacks accurately.

5. Conclusion
This paper proposed a kind of edge security control mechanism based on intelligent whitelist verification, as a method for monitoring abnormal network access at the network edge and microboundary. Through the intelligent detection and analysis of the security feature data of the whitelist permission items, the goal to control the network edge security has been proposed with the intelligent whitelist. Finally, through comparative experiments, the edge security control mechanism proposed has been proved to be effective to detect the abnormality of network access. In the future, we will verify the real-time performance of the method in practical applications in large-scale networks and optimize the self-learning mechanism and capabilities of the whitelist.

Data Availability
The data included in this paper are available upon request to the corresponding author without any restriction.

Conflicts of Interest
The authors declare that they have no conflicts of interest regarding the publication of this paper.

References
[1] R. Mitchell and I.-R. Chen, “A survey of intrusion detection techniques for cyber-physical systems,” ACM Computing Surveys, vol. 46, no. 4, pp. 1–29, 2014.
[2] R. Singh, H. Kumar, R. K. Singla, and R. R. Ketti, “Internet attacks and intrusion detection system,” Online Information Review, vol. 41, no. 2, pp. 171–184, 2017.
[3] H.-J. Liao, C.-H. Richard Lin, Y.-C. Lin, and K.-Y. Tung, “Intrusion detection system: a comprehensive review,” Journal of Network and Computer Applications, vol. 36, no. 1, pp. 16–24, 2013.
[4] D. Kapil, N. Mehra, A. Gupta, S. Maurya, and A. Sharma, “Network security: threat model, attacks, and IDS using machine learning,” in Proceedings of the 2021 International Conference on Artificial Intelligence and Smart Systems (ICAISS), Coimbatore, India, March 2021.
[5] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep learning for anomaly detection," ACM Computing Surveys, vol. 54, no. 2, pp. 1–38, 2021.
[6] T. Saranya, S. Sridevi, C. Deisy, T. D. Chung, and M. K. A. A. Khan, "Performance analysis of machine learning algorithms in intrusion detection system: a review," Procedia Computer Science, vol. 171, pp. 1251–1260, 2020.
[7] W. Liu, P. Ren, K. Liu, and H. X. Duan, Intrusion Detection Using SVM, pp. 1–4, IEEE, Mountain, NY, USA, 2011.
[8] I. Gumus and Y. Sirin, “High-dimensional and wide-scale anomaly detection using enhancing support vector machine,” in Proceedings of the 2018 26th Signal Processing and Communications Applications Conference (SIU), Izmir, Turkey, May 2018.
[9] C. Kruegel, D. Mutz, W. Robertson, and F. Valeur, "Bayesian event classification for intrusion detection," in Proceedings of the Computer Security Applications Conference, IEEE, Austin, TX, USA, December 2010.

[10] S. Vargaftik, I. Keslassy, A. Orda, and Y. Ben-Itzhak, "RADE: resource-efficient supervised anomaly detection using decision tree-based ensemble methods," Machine Learning, vol. 110, no. 10, pp. 2835–2866, 2021.

[11] S. Sivanantham, R. Abirami, and R. Gowsalya, "Comparing the performance of adaptive boosted classifiers in anomaly based intrusion detection system for networks," in Proceedings of the 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), Vellore, TN, India, March 2019.

[12] K. Cho, A. Courville, and Y. Bengio, "Describing multimedia content using attention-based encoder-decoder networks." IEEE Transactions on Multimedia, vol. 17, no. 11, pp. 1875–1886, 2015.

[13] G. Liu and J. Zhang, "CNID: research of network intrusion detection based on convolutional neural network," Discrete Dynamics in Nature and Society, vol. 2020, Article ID 4705982, 11 pages, 2020.

[14] T. Ergen and S. S. Kozat, "Unsupervised anomaly detection with LSTM neural networks," IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 8, pp. 3127–3141, 2020.

[15] O. Atkinson, A. Bhardwaj, C. Englert, V. S. Ngairangbam, and M. Spannowsky, "Anomaly detection with convolutional graph neural networks," Journal of High Energy Physics, vol. 2021, no. 8, 19 pages, Article ID 4705982, 2021.

[16] A. Raj, M. Poriye, and M. Poriye, "Anomaly-based intrusion detection system using supervised learning algorithm artificial neural network and ant colony optimization with feature selection," International Journal of Engineering and Advanced Technology, vol. 9, no. 3, pp. 2475–2481, 2020.

[17] I. S. Thaseen, C. A. Kumar, and A. Ahmad, "Integrated intrusion detection model using chi-square feature selection and ensemble of classifiers," Arabian Journal for Science and Engineering, vol. 44, no. 4, pp. 3357–3368, 2018.

[18] S. Waskle, L. Parashar, and U. Singh, "Intrusion detection system using PCA with random forest approach," in Proceedings of the 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 803–808, Coimbatore, India, July 2020.

[19] C. Callegari, S. Giordano, M. Pagano, and M. Pagano, "An information-theoretic method for the detection of anomalies in network traffic," Computers & Security, vol. 70, pp. 351–365, 2017.

[20] M. Ahmed, A. Naser Mahmood, and J. Hu, "A survey of network anomaly detection techniques," Journal of Network and Computer Applications, vol. 60, pp. 19–31, 2016.

[21] Y. Hou, S. Garg, L. Hui, D. N. K. Jayakody, R. Jin, and M. S. Hossain, "A data security enhanced access control mechanism in mobile edge computing," IEEE Access, vol. 8, pp. 136119–136130, 2020.

[22] M. Manavi and Y. Zhang, "A new intrusion detection system based on gated recurrent unit (GRU) and genetic algorithm," Security, Privacy, and Anonymity in Computation, Communication, and Storage, Springer, Berlin, Germany, pp. 368–383, 2019.