Research of Network Intrusion Detection Based on Improved Seagull Optimization Algorithm with Deep Learning

Hai Lan(✉)

Wuhan FiberHome Information Integration Technologies Co., Ltd., Wuhan, China
lanhai@fiberhome.com

Abstract. In this paper, we study a network intrusion detection method based on deep learning combined with improved seagull optimization algorithm, which extracts the information traces inevitably generated during network intrusion by deep neural network and optimizes the parameters of deep neural network model by improved seagull optimization algorithm, so as to build an efficient network intrusion detection model. The traditional seagull optimization algorithm is improved and applied to the deep learning model hyperparameter optimization. For the shortcomings of the traditional seagull optimization algorithm with strong randomness of population initialization and easy to produce extreme individuals, a reverse learning method is presented to the initialization of the group. And a nonlinear convergence factor is used to enhance the convergence speed, thus improving the performance of the seagull optimization algorithm. The improved algorithm was demonstrated by using standard test functions, and the improved algorithm was used for parameter optimization of the deep learning model. To address the shortcomings of classical rule-based, host behavior analysis, and machine learning network traffic classification methods in performing network intrusion detection with less attention to the temporal correlation characteristics of samples, we propose to apply deep learning techniques to network intrusion detection, and design a network intrusion detection method based on gated cyclic units and multilayer perceptron, and also apply the improved seagull optimization algorithm to the method optimization of hyperparameters in the model, thus improving the performance of the model and achieving better network intrusion detection results on the NSK-KDD dataset.

Keywords: Network intrusion detection · seagull optimization algorithm (SOA) · deep learning · gated recurrent unit (GRU)

1 Introduction

In terms of research content in the field of network security, network intrusion detection is a very important issue that needs extensive attention, and it is also a prerequisite for ensuring network security. On the one hand, it is necessary for companies, enterprises and individuals to detect network intrusions to prevent possible network intrusions.

W. Hong and G. Kanaparan (Eds.): ICCSE 2023, CCIS 2023, pp. 100–112, 2024.
https://doi.org/10.1007/978-981-97-0730-0_10

Losses due to damage to network equipment. On the other hand, cyberspace is also the sovereign space of a country, which is open and inviolable at the same time. Sovereign countries first need to realize the detection of network intrusion from a macro level, and grasp the trend of network intrusion crimes. Timely discover network intrusions that affect national security and social and public interests, and have countermeasures in the face of attacks such as network intrusions, so as to avoid cyberspace becoming an "extra-legal place" and safeguard network sovereignty [4].

However, the rapid development of the Internet also means that the network structure is more complex and the network traffic continues to grow, which means that network intrusion detection faces more and greater challenges. Faced with the current situation, some traditional network intrusion detection technologies have become inadequate, and their shortcomings are mainly concentrated in the following points: First, in the era of big data, the amount of intrusion data is large, and traditional network intrusion detection technologies face such situations. Unable to handle and cope. Second, with the continuous advancement of Internet technology, the attack methods emerge in an endless stream, and traditional rule matching and host-based network intrusion detection methods are difficult to deal with. Third, some new attack methods continue to emerge, which also leads to the problem of imbalance between old and new intrusion data, and traditional network intrusion detection algorithms are inefficient. It is because of these drawbacks that the traditional network intrusion detection technology is difficult to withstand the current network intrusion, so the existing technology needs to be innovated, and deep learning is a method that can change this situation. Deep learning is one of the hotspots of research in recent years. This technology has been applied to many fields, especially in the fields of Computer Vision (CV) and Natural Language Processing (NLP) [5]. For network intrusion detection, deep learning is more intelligent. It can automatically learn to extract and store the spatiotemporal features of data, and continuously learn and strengthen in the process of model building and training. In terms of process, deep learning is an end-to-end process. Learning, data collection, input, processing, and output are systematic, convenient and fast. Therefore, it is a reasonable and effective method to apply depth to network intrusion detection. It is necessary and practical to study how to use deep learning technology for network intrusion detection.

The research on network intrusion detection can be traced back to the 1980s. In 1980, Anderson put forward a famous computer security threat model in a technical report titled "Computer Security Threat Monitoring and Monitoring", and this threat is today's intrusion in a broad sense [1]. In the technical report, intrusion is classified into three types: external intrusion, internal penetration and abuse, and expounds the method of tracking these intrusion activities through audit data, creating a precedent for network intrusion detection. The design and development of intrusion detection system (IDS) by later generations basically take this security threat model as the basic starting point. In 1987, Denning proposed a model of a general intrusion detection system, which brought intrusion detection system theory to the computer network security system [6]. The model proposed by Denning divides intrusion detection into six parts: subject, object, audit record, activity filing, exception record, and activity rule. This model is also considered to be the first substantial intrusion detection system prototype. On the basis of this model, the research on intrusion detection has also begun to be widely carried

out. It is worth mentioning that in 1990, Heberlein and others from The University of California Davis put forward the concept of Network Security Monitor (NSM). NSM system takes network traffic packets as the information source to detect intrusion for the first time, which is also the first time that network-based intrusion detection system is proposed [7], which also affects the development of subsequent intrusion detection technology. After that, as a rising star, compared with the original method based on rule matching and host analysis, network-based intrusion detection technology has become more and more sought after. In terms of the difference in several ways, the host-based intrusion detection system collects the required data from its host device, and uses computer log files as important analysis data. Compare and analyze to achieve the effect of detection. The advantages of the host-based intrusion detection system are that the equipment loss and personnel operation cost are low, and the false alarm rate is relatively low. Limited and complex. The network-based intrusion detection system is inseparable from the data packets communicated by network devices. It realizes the determination of network intrusion behaviors by collecting, encoding, decoding, and analyzing network data packets, which is more flexible and convenient. The method based on rule matching requires the establishment and update of a rule database, which is inefficient and has been gradually eliminated (Fig. 1).

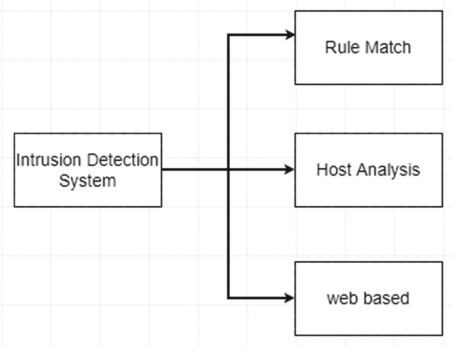


Fig. 1. Intrusion detection system classification

With the development of The Times and technological progress, machine learning technology has attracted the attention of many scholars, and a large number of scholars have participated in the research on how to apply machine learning to network intrusion detection. Traditional machine learning methods have been widely used in network intrusion detection [8, 9]. The intrusion detection system designed by Saleh et al. is a mixture of machine learning methods, which is real-time and can solve multiple classification problems. In data processing, dimensionality reduction is an important operation, this system uses the basis feature selection (NBFS) technology to first process the data samples, and then eliminates the samples with high dispersion through the optimized support vector machine (OSVM). Finally, PKNN (the improved version of classical KNN) is used to detect attacks. Experiments were carried out on classic KDD 99, NSL-KDD and Kyoto2006 + data sets to demonstrate the real-time performance of detection and achieve

good detection results [10]. In literature [11], support vector machine (SVM) integrated with compressed sampling is used for intrusion detection, and compression sampling method is used for network data flow processing. A SVM-based model is established to output results, and good experimental results have been achieved in training efficiency and detection speed. Literature [12] proposed an ingenious data processing method, which used Fisher discriminant ratio (FDR) method for feature extraction and denoising. In the subsequent processing, Probabilistic self-organizing Maps (PSOM) was used for modeling to preserve the characteristics of data, which achieved a good effect on classification accuracy. The training time of classifier is also an indicator worthy of attention. Al-Yaseen et al. constructed a multi-level model to improve the training efficiency and the performance of the model. First of all, in terms of data processing, the improved K-means algorithm is used to purify and sample the original training set. Starting from the specific data itself, the optimized data training efficiency is greatly improved. In the final result output stage of the model, the combination of extreme learning machine (ELM) and support vector machine was used to carry out multi-classification, and the classical KDD99 data set was used for experimental evaluation, and the accuracy rate reached 95.75% [13]. See from the above research achievements of many, in machine learning method is applied to solve the problem of network intrusion, scholars generally will first some efficient methods on data processing (e.g., sampling), and then by machine learning methods to realize the feature extraction, build the model for classification or prediction and the final result output. In the selection of models, we can either stack the basic models, or improve the basic model, or combine the two. The ultimate goal is nothing more than to build an efficient model. However, today's network attack traffic keeps increasing, data dimensions keep increasing, attack methods keep innovating, and the seriousness of data imbalance becomes increasingly prominent. Traditional machine learning methods are also unable to cope with these complex situations. First of all, it is the feature extraction. Manual plays an important role in the process of feature extraction, but it is difficult for manual to explore the internal features of data. It mainly relies on experience to operate, unable to distinguish the correlation of data and temporal and spatial characteristics well, and difficult to explore the internal connections and rules of data. Moreover, from the perspective of data volume, such methods are obviously inefficient in the face of big data. Therefore, the intrusion detection technology of traditional machine learning method is faced with many difficulties, and new technology and new ideas are in urgent need of development.

2 Network Intrusion Detection Method Based on Improved Seagull Optimization Algorithm and Gated Recurrent Unit

2.1 Improved Seagull Optimization Algorithm

2.1.1 Inverse Learning and Nonlinear Convergence Factors

When the traditional seagull algorithm is used for optimization, the initial seagull group has strong randomness, the optimization efficiency is low, and it is easy to fall into the local optimum. Because when the traditional seagull optimization algorithm is initialized, the population individuals are randomly generated, and some extreme individuals

are prone to appear, thus affecting the initialization quality of the population, which has a certain negative impact on the performance of the algorithm. The method based on reverse learning can improve the quality of the initialized population.

The concept of reverse exists not only in each of our consciousness, but also plays a special role in the natural world and exists in different ways. For example, opposite particles in high-energy physics, antonyms in language and literature, absolute or relative complement, dual variables in mathematics, subject and object in philosophy, etc. The basic reverse concept first appeared in the ancient Chinese symbol of yin and yang, which embodies the dualistic concept that black is yin and white is yang. In addition, the natural pattern in Greek classical philosophy is also closely related to the reverse, such as fire to water, heat to cold, dry to wet, earth to air, and many entities or situations can be described by the concept of reverse.

In fact, the interpretation of different entities can be made easier using the reverse concept. And paired oppositions like east, west, south, and north cannot be defined individually, only when they are used together can they be explained to each other. Therefore, Tizhoosh is inspired by the concept of reverse in the real world, and proposes the concept of reverse in computation, that is, reverse-based learning, referred to as reverse learning [18].

In ISOA, the population position initialization process based on the reverse learning method is expressed as follows (N represents the population size of seagulls, and t is the current number of iterations):

(1) Randomly initialize the population:

$$P(t=0) = \left\{x_{m,n}\right\} 1, 2, \ldots, N;\ n = 1, 2, \ldots N \tag{1}$$

(2) Calculate the reverse population through the formula:

$$P'(t=0) = \left\{x'_{m,n}\right\},\ x'_{m,n} = x_{min,n} + x_{max,n} - x_{m,n} \tag{2}$$

Calculate the reverse population, where $x_{min,n}$ and $x_{max,n}$ are the worst and best individuals, respectively.

(3) Select the top N good individuals: select the top N individuals with the best fitness from the set $\{P(t=0)\} \bigcup \left\{P^{'}(t=0)\right\}$ as the initialization population.

As mentioned above, we obtained the initialization population with higher fitness through the reverse learning method. In the iterative process of the algorithm, the movement tendency of the individual population significantly affects the convergence speed of the algorithm. From formula (2), it is not difficult to see that in the traditional seagull optimization algorithm, the convergence factor changes linearly, which leads to poor performance and low efficiency in the later stage of the algorithm. This paper proposes a new convergence factor formula, which is described as follows:

$$A = f_c\left(1 - \frac{1}{e-1} \times \left(\frac{t}{e^{Max_{iteration}}} - 1\right)\right) \tag{3}$$

In the above formula, f_c represents the initial convergence value (usually we set it to 2), t represents the number of iterations of the current population, and $Max_{iteration}$

represents the maximum number of iterations of the population we set. After applying this formula, the seagull optimization algorithm has slow convergence in the early stage and fast convergence in the later stage, so that the global search ability in the early stage and the convergence ability in the later stage can be improved while maintaining the diversity of the population.

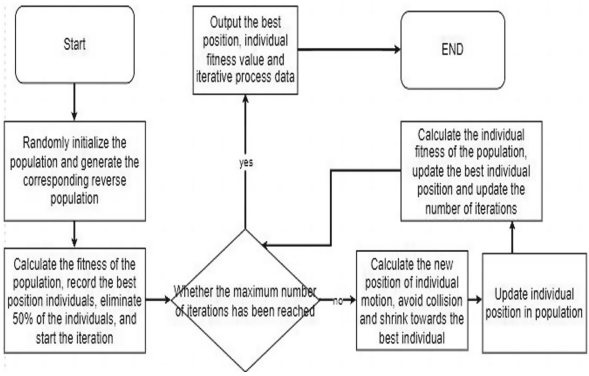
In the next two sections, we will show the above-mentioned Improved Seagull Optimization Algorithm (ISOA) process, and verify the performance of our improved algorithm through comparative experiments.

2.1.2 Improved Seagull Optimization Algorithm Flow

The improved Seagull Optimization Algorithm (ISOA) proposed in this paper adopts the population initialization strategy of reverse learning and nonlinear convergence factor to improve the convergence speed and global optimization ability of the algorithm. The steps of the improved ISOA algorithm are as follows:

(1) Population initialization. First, set the population size, the limit of the number of iterations of the algorithm, the movement range of the population, the initial value of the convergence factor, and the movement constants u, v.
(2) Generate a reverse population and select the best. Use the reverse learning algorithm to generate the reverse population, calculate the fitness value of each individual seagull, find the optimal value (optimal seagull position) of the entire population, and eliminate the last 50% of the individuals.
(3) Iterative optimization. Within the set iteration range, according to formulas (5)–(10), calculate the position to which the individual will move, and use formulas (2) and (3) to avoid individual collision and continuous shrinkage.
(4) Update the population. Calculate the fitness value of the new position of the individual, and determine whether the individual needs to move through the comparison, so as to update the population position and update the best individual (position).
(5) Repeat the process of (3) and (4) until the iteration ends.
(6) Output the best individual (position), fitness value and iterative process record of the seagull population, and the algorithm ends.

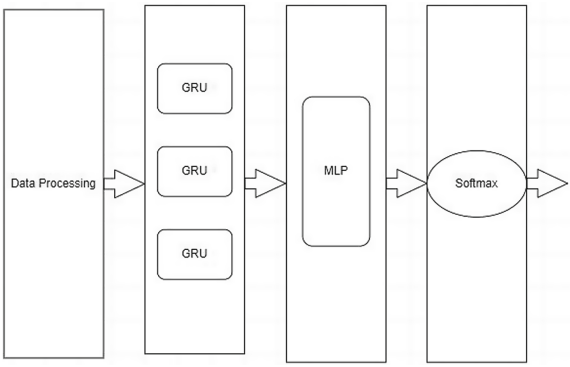
The corresponding flowchart is as follows:



Flow chart of ISOA algorithm

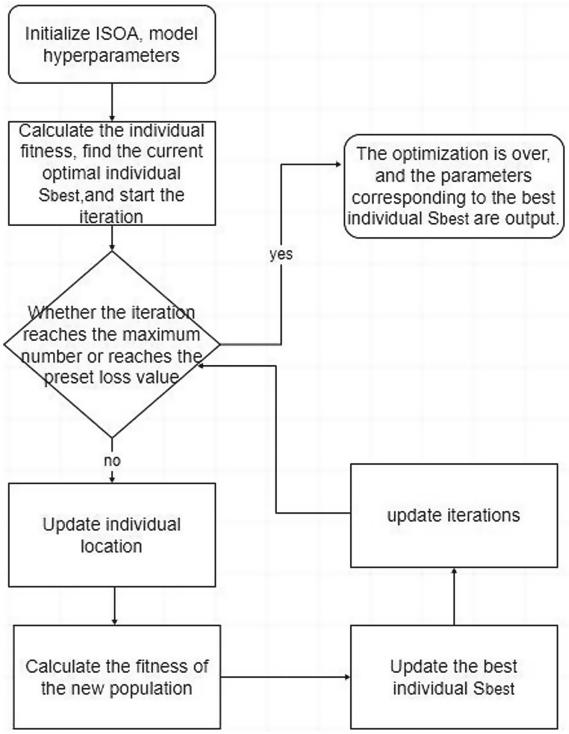
2.1.3 Network Intrusion Detection Method Based on ISOA and GRU

In order to build a deep neural network model for network intrusion detection, GRU, MLP and other components can be regarded as independent modules of the network, and multiple modules are in a cascade relationship. By combining these modules, the overall structure of our proposed intrusion detection model is shown in the following figure.



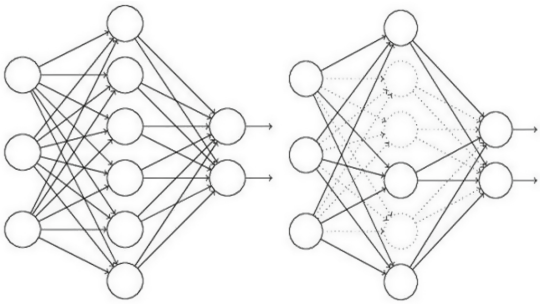
Schematic diagram of model structure

The system consists of data processing module, GRU module, MLP module and output module. The data processing module processes the data into normalized values suitable for input to the neural network without changing the dimension of the data. The setting of some parameters of the model is optimized by the ISOA algorithm. By setting some parameters of the model as the activity range of the population in the ISOA algorithm, and using the loss function in the model as the population fitness calculation formula, the parameters are encoded and set as individual. First, set the parameters and initialize the population, and then access the training module. After obtaining the training fitness, use ISOA to update the individual. The update process will retain the best parameters for comparison. When the number of iterations reaches the limit or the fitness threshold appears during the training process, we will end the process and output the best parameter combination. In order to improve the efficiency of optimization, we only take 20% of the data sets in NSL-KDD as training data to optimize parameters. The overall process is as follows:



Schematic diagram of parameter flow of ISOA optimization model

The GRU module is composed of one or more layers of GRUs. For the input data, it is mainly used to extract and store features, and is the core part of the system. The MLP module is an n-layer perceptron model, which mainly performs nonlinear mapping on the output information of the GRU again to realize classification decisions. The output module uses Softmax regression to normalize the final classification probability output. In order to improve the phenomenon of model overfitting, we use the dropout algorithm. As shown in Fig. 8, dropout is an important layer in the actual experimental processing. The method is to discard some neurons. The proportion of discarding depends on the experimental experience. Numerically speaking, it is a proportional value. During the training process, A discarded node means that the value of the node is 0. The training of each batch is repeated, and after some neurons are randomly discarded, the interaction between neurons will be alleviated, preventing model overfitting and improving model performance.



Schematic diagram of dropout process

2.2 Experiment and Analysis

2.2.1 Data Processing and Experimental Design

Since the proposed detection system can only accept numerical input, it is first necessary to convert the different categories in the dataset into numerical data that can be input into the neural network. In each record, only 3 features (protocol_type, service and flag) are symbolic features that need to be converted to numerical data. This is achieved here with 1 to N encoding. Similarly, the classification results are represented by numbers 0~4, as shown in Table 1. Then all the numerical features need to be normalized, that is, scaled between 0 and 1. Here, min-max normalization is used to complete the linear scaling, and the formula is shown in formula (4).

$$f' = \frac{f - min_j}{max_j - min_j} \tag{4}$$

f represents the original value of the feature, and f' represents the normalized feature value. $max_j - min_j$ are the maximum and minimum values of the j th feature.

Table 1. Label value conversion

Label	Numerical encoding
NORMAL	0
DOS	1
R2L	2
U2R	3
PROBING	4

The software and hardware environment of the experiment is still the same as that described in Sect. 2, and the framework is used to facilitate us to quickly build the model. The process of data processing is also a part of our intrusion detection system. After

processing, the data will be sent to our model for training, and the final result will be output. In the model part, the GRU module and the MLP module are undoubtedly the most important. We will use the ISOA optimization combination model method named ISOA + GRU + MLP model. In the model comparison part, we use separate MLP, GRU models, and GRU + MLP combined models to conduct experiments.

Among them, starting from our experimental experience, the ISOA + GRU + MLP model hyperparameter optimization includes: batch size, learning rate, number of MLP layers, and number of MLP hidden layer units. The range of batch size we set is [10, 50]; the range of learning rate is [0.005, 0.2]; the range of MLP layers is [1, 5]; the range of MLP hidden layer units is [32, 128]. After the optimization process described in the previous section is optimized, the values are as follows (Table 2):

Table 2. Model Hyperparameter Settings

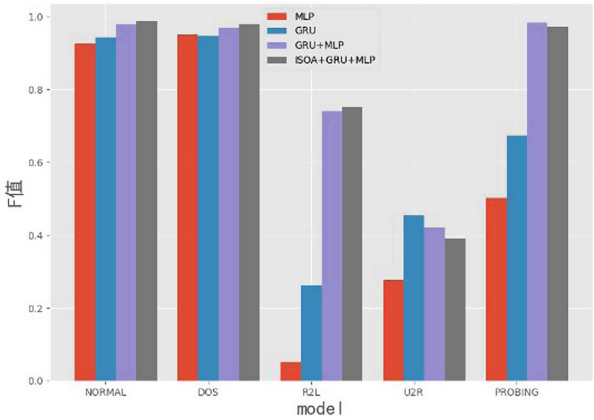
Hyperparameter	Value
Batch size	30
Learning rate	0.01
Number of MLP layers	3
Number of MLP hidden layer units	65

2.2.2 Experimental Results and Analysis

In order to objectively evaluate the performance of the method proposed in this chapter, we take 80% of the data set as the training set and 20% as the test set, the number of runs is 20 times, and the results are averaged. The experimental results are as follows:

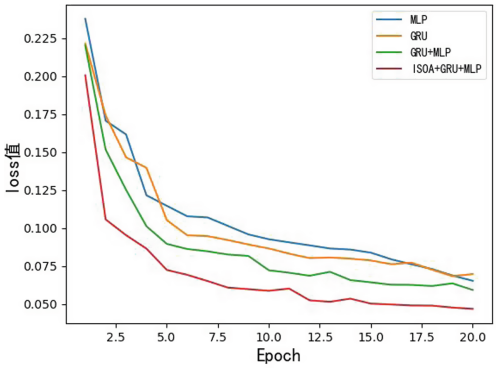
Table 3. Test results of different models

Dataset	Algorithm model	Average accuracy(ACC)	Best accuracy (BACC)	Detection rate(DR)	False alarm rate (FAR)
NSL-KDD	MLP	91.79%	93.34%	90.31%	4.36%
	GRU	94.91%	96.46%	90.75%	9.35%
	GRU + MLP	96.13%	98.35%	95.41%	3.09%
	ISOA + GRU + MLP	**97.59%**	**98.61%**	**97.94%**	**2.01%**



F-value results on NSL-KDD dataset

As can be seen from the results in Table 3, in the NSL-KDD dataset, the results of ISOA + GRU + MLP are the best. In terms of average accuracy, the performance rankings of the models are: ISOA + GRU + MLP > GRU + MLP > GRU > MLP, the average accuracy of ISOA + GRU + MLP reached 97.59%, the best accuracy reached 98.61%, the average accuracy was 1.46% higher than that of GRU + MLP, and the detection rate was also the highest among the five models. The false alarm rate is as low as 2.01%, which proves that the model we designed has excellent performance. As can be seen from Fig. 9, in terms of attack types, the comparison of detection rate and F value shows that the detection effect of DOS attacks and PROBING attacks is significantly better than that of R2L and U2R attacks.



Variation of loss value in different models

Figure 10 shows the variation curve of the experimental average loss value, and compares the convergence of different algorithms. It can be seen that the ISOA + Gru + MLP model proposed by us has fast convergence speed and lower loss value. Combined with the previous results, we can see that using the ISOA + Gru + MLP model not only has advantages in accuracy, but also has a fast convergence speed.

3 Conclusion

In this chapter, a network intrusion detection method based on the improved seagull optimization algorithm and gated recurrent unit is proposed, and the system framework of the network intrusion detection method based on the improved seagull optimization algorithm and the gated recurrent unit is described in detail. We combine a recurrent neural network with a gated recurrent unit and a multi-layer perceptron to construct an intrusion detection system based on a deep neural network, and use the deep learning method to train, and achieve good intrusion detection performance. In the experimental part, the commonly used network intrusion detection data sets are firstly introduced, the advantages of the data characteristics of the NSL-KDD data set are highlighted, and the various evaluation indicators and key points of the experiment are described in detail. The experimental comparison of various combinations shows that the ISOA optimization GRU and MLP proposed in this paper have the best combination performance, excellent performance indicators, and the highest detection accuracy rate of 98.61%, especially in the DOS attack type the average accuracy rate reached 99.17%.

Acknowledgment. This work is funded by the National Natural Science Foundation of China under Grant No. 61772180, the Key R & D plan of Hubei Province (2020BHB004, 2020BAB012).

References

1. Anderson, J.P.: Computer security threat monitoring and surveillance. Technical report, James P. Anderson Company (1980)
2. Sherstinsky, A.: Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Physica D **404**, 132306 (2020)
3. Chung, J., Gulcehre, C., Cho, K.H., et al.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 (2014)
4. Hong, Y., Goodnight, G.T.: How to think about cyber sovereignty: the case of China. Chin. J. Commun. **13**(1), 8–26 (2020)
5. Coşkun, M., Yildirim, Ö., Ayşegül, U., et al.: An overview of popular deep learning methods. Eur. J. Tech. (EJT) **7**(2), 165–176 (2017)
6. Denning, D.E.: An intrusion-detection model. IEEE Trans. Software Eng. **2**, 222–232 (1987)
7. Heberlein, L.T., Dias, G.V., Levitt, K.N., et al.: A network security monitor. Lawrence Livermore National Lab., CA (USA); California Univ., Davis, CA (USA). Dept. of Electrical Engineering and Computer Science (1989)
8. Sommer, R., Paxson, V.: Outside the closed world: on using machine learning for network intrusion detection. In: 2010 IEEE Symposium on Security and Privacy, pp. 305–316. IEEE (2010)
9. Song, W., Beshley, M., Przystupa, K., et al.: A software deep packet inspection system for network traffic analysis and anomaly detection. Sensors **20**(6), 1637 (2020)
10. Saleh, A.I., Talaat, F.M., Labib, L.M.: A hybrid intrusion detection system (HIDS) based on prioritized k-nearest neighbors and optimized SVM classifiers. Artif. Intell. Rev. **51**(3), 403–443 (2019)
11. Chen, S., Peng, M., Xiong, H., Yu, X.: SVM intrusion detection model based on compressed sampling. J. Electr. Comput. Eng. **2016**, 1–6 (2016)

12. De La Hoz, E., Ortiz, A., Ortega, J., Prieto, B.: PCA filtering and probabilistic SOM for network intrusion detection. Neurocomputing **164**, 71–81 (2015)
13. Xu, H., Przystupa, K., Fang, C., Marciniak, A., et al.: A combination strategy of feature selection based on an integrated optimization algorithm and weighted k-nearest neighbor to improve the performance of network intrusion detection. Electronics **9**(8), 1206 (2020)