Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105 (Online)
Volume **10**, Number **2**, May **2025**

Intrusion Detection Algorithm Based on Fusion Feature Selection in Intelligent Measurement Systems

Peng Li

Joint Laboratory of Digital Technology for New Power System
Digital Grid Research Institute China Southern Power Grid, Guangzhou 510700, China
lipeng@csg.cn

Zhi-Ming Wang*

Guangdong Provincial Key Laboratory of Digital Grid Technology
Digital Grid Research Institute China Southern Power Grid, Guangzhou 510700, China
jamewzm@163.com

Zhao-Hui Hu

Guangdong Provincial Key Laboratory of Digital Grid Technology
Southern Power Grid Digital Platform Technology (Guangdong) Co., Ltd, Guangzhou 510700, China
Huzh@csg.cn

Wei-Xun Zhang

Joint Laboratory of Digital Technology for New Power System
Digital Grid Research Institute China Southern Power Grid, Guangzhou 510700, China
zhangwx5@csg.cn

Jian Ma

Guangdong Provincial Key Laboratory of Digital Grid Technology
Digital Grid Research Institute China Southern Power Grid, Guangzhou 510700, China
352236881@qq.com

Ting-Wen Yu

Guangdong Provincial Key Laboratory of Digital Grid Technology
Southern Power Grid Digital Platform Technology (Guangdong) Co., Ltd, Guangzhou 510700, China
ytw@csg.cn

*Corresponding author: Zhi-Ming Wang

Received January 28, 2024; revised August 1, 2024; accepted November 20, 2024.

Abstract. *The existing intrusion detection algorithms are difficult to completely abstract the features contained in the intrusion behavior. This paper aims at improving the efficiency of intrusion detection based on the characteristics that intrusion detection data has many redundant features, Conduct in-depth research around feature selection method and model parameter synchronization optimization. For the model method proposed in this paper, NSL-KDD data set is used for experimental verification, and compared with the existing perceptual learning algorithm, logical regression algorithm, neural network algorithm, and deep learning algorithm. The experimental results show that the proposed GWO-DF intrusion detection algorithm has good effect in feature selection and detection efficiency.*
Keywords: Intrusion detection, Deep learning, Recursive feature elimination, Deep forest, Grey Wolf optimization algorithm.

1. **Introduction.** Intrusion Detection is a solution to security problems, which can detect abnormal behaviors in the system in time and immediately notify the computer to carry out corresponding security protection work, so as to achieve the goal of protecting computer systems and information.

Nowadays, artificial intelligence technology continues to expand to various fields, and researchers have also applied artificial intelligence technologies such as machine learning to the field of abnormal intrusion detection [1], and proposed a new attack detection method, through the analysis and processing of massive data, to discover the essential information of things, and then enable computers to obtain the ability to simulate human behavior, which has become an extremely important technology in network intrusion detection. For example, intrusion detection methods based on machine learning such as Decision Tree (DT) [2], Support Vector Machine (SVM) [3], and Naive Bayesian (NB) [4] have been gradually proposed and used.

Liang et al. proposed an industrial network intrusion detection algorithm based on multi-feature data clustering optimization model, which improved the detection rate and real-time performance of abnormal behavior detection of multi-feature data in industrial networks [5]. He et al. proposed a wrapping feature selection method for the eXtreme Gradient Boosting (XGBoost) algorithm, which not only simplifies the dataset but highlights the importance of different features in intrusion detection, and improves the efficiency of intrusion detection [6]. Zhang et al. proposed an industrial control network intrusion detection technology based on semi-supervised machine learning [7], which improved the detection ability of abnormal traffic to a certain extent. Liu et al. proposed a distributed and collaborative network intrusion detection technology, which can be better applied to network intrusion detection technology in the context of big data [8]. Wang et al. proposed an improved optimization algorithm to optimize the intrusion detection model of SVM industrial control system [9] to improve the detection accuracy and detection speed of the industrial control intrusion detection model. Kundu et al. proposed automatic machine malware detection, and applied the technology of automatic machine learning to examine the model construction of intrusion detection from a new perspective, which can identify the set of hyperparameters with significantly better performance than the model performance under the best known hyperparameter settings [10, 11], and the recognition effect of the model is also significantly improved.

In the process of model learning, data feature selection is the key to improve the performance of machine learning model, but the process of feature selection requires manual intervention and has a certain artificial dependence. Nowadays, with the advent of the era of big data, network traffic often has higher-dimensional characteristics, so how to pick out the data features with greater improvement on the model in multi-dimensional features is the main problem, and the deep learning model can automatically extract the data sample features and reduce the dependence on manual labor, so more and more scholars are applying the deep learning to the field of intrusion detection, and deep learning become a research hotspot in the field of artificial intelligence [12]. Deep learning algorithms such as Convolutional Neural Network (CNN) [13], Recurrent Neural Network (RNN) [14], and auto-encoder [15] have been applied to intrusion detection systems and have achieved certain results. Liu et al. fused CNN and Bi-directional Long Short-Term Memory Bi-directional Long Short-Term Memory (Bi-LSTM) to construct a network intrusion detection model [16], realized a convolutional neural network with a double-layer structure, and achieved good results. Terai et al. proposed an intrusion detection method

based on the Long Short Term Memory (LSTM) model, which takes less time to detect attack packets [17].

Although the development of intrusion detection technology is relatively rapid, with the development of intrusion detection technology, the means of network attacks are also constantly updated, and with the deepening of the integration of life and the Internet, the network environment has become more complex [18], some studies have shown that the ensemble learning algorithm can further improve the performance of the classification algorithm on the basis of a single deep learning algorithm or machine learning algorithm, and this paper proposes a Grey Wolf Optimizer-Deep forest (GWO-DF) with fusion feature selection Optimizer-Deep forest algorithm, in which Recursive Feature Elimination (RFE) is used in the feature selection method [19], which can reduce the feature dimension to avoid the dimensionality disaster, so as to select the best feature combination and improve the efficiency of the classification model, and find the optimal model through the gray wolf optimization algorithm to improve the efficiency and accuracy of detection [20].

2. **Network Intrusion Detection Dataset.**

2.1. **NSL-KDD dataset.** The NSL-KDD dataset is used as the benchmark dataset, which is an improved version of the KDD-CUP99 dataset, which not only effectively solves the inherent redundant record problem of the KDD-CUP99 dataset, but also makes the number of records in the training set and the test set reasonable, so that the classifier is no longer biased towards more frequent records.

The dataset includes the KDD train+ dataset as the training set and the KDD test+ and KDD test21 datasets as the test set, and the attacks in the dataset are divided into four types: denial-of-service attack (DoS), root to local user attack (R2L), common user to root attack (U2L), and probe attack (Prob).

Normal: Indicates a normal network connection.

DoS attack: A denial-of-service attack in which extraordinary visitors use all methods to paralyze a service or network, causing the target machine to be unable to provide services to users, such as increasing bandwidth consumption. Attacks of this type in the dataset are identified as mailbomb, smurf, udpstorm, apache2, etc.

R2L attack: An attacker uses remote access to threaten computer data privacy. Specifically, there are guess_passwd, ftp_write, multihop, and spy.

Probe Attack: Port scanning or monitoring, the attack uses programs or tools to continuously scan the host port to find vulnerabilities, which generally does not pose a threat to the host, and mainly collects vulnerability information for the next attack. The dataset mainly includes six types of port scanning attacks, including ipsweep, nmap, portsweep, satan, mscan, and saint.

U2R: An unauthorized user obtains the highest administrative privilege as root, and attacks use vulnerabilities to directly obtain the root privilege or the highest privilege to control the host and perform unauthorized operations. The specific attack types include buffer_overflow, loadmodule, perl, and rootkit.

Table 1 shows the number of normal records and four different types of attack records in each dataset.

Each traffic record in the NSL-KDD dataset has 41 characteristics: 34 continuous values and 7 discrete values.

2.2. **Data Preprocessing.** Therefore, the purpose of this paper is to convert the character features with realistic significance in the dataset into digital features that can be used in the subsequent learning model, and the preprocessing of the dataset is divided into two

TABLE 1. Number of different attack types in the dataset

Type	KDDTrain	KDDTest	KDDTEST-21
Normal	67343	9711	2152
DoS	45927	7458	4342
Probe	11656	2421	2402
R2L	995	2754	2754
U2R	52	200	200
Total	125973	22544	11850

steps: (1) the one-hot coding method is used to transform the data of the dataset. (2) data standardization.

(1) One-hot coding: the one-hot coding is used because the subsequent model algorithm is calculated based on the measure in the vector space, in order to make the variable values of the non-partial order relationship not partial order, and the distance to the origin is equal, the one-hot encoding is used to convert the value of the character feature into a numeric feature vector, and the value of the discrete feature is extended to the Euclidean space, and a certain value in the character feature corresponds to a certain point in the Euclidean distance space. Makes the distance between features more reasonable. For example, a feature protocol_type has three eigenvalues: TCP, UDP, and ICMP, which are encoded and converted into binary vectors of (1,0,0), (0,1,0), and (0,0,1). Similarly, service has 70 attribute features, flag has 11 attributes, etc. The one-hot encoding method is used to process such character features, and the 41-dimensional feature vectors in the final dataset are finally converted into 122-dimensional feature vectors.

(2) Data normalization: The purpose of data normalization is that the dimensions between different features in the data are inconsistent, and the difference between the maximum and minimum values of some features in the data is very large, for example, the duration feature in the dataset has a value range of [0,58329], while the value range of the src_bytes features in the dataset is [0,1.3*109] and the value range of the dst_bytes feature is [0,1.3*109]. If it is not processed, the model training results will be affected, so the data is scaled to a certain scale so that it falls into a feature area for comprehensive analysis. In this paper, the maximum-minimum normalization method is used, as shown in Equation (1).

$$\overline{x_i} = \frac{x_i - Min}{Max - Min} \tag{1}$$

Each feature is linearly mapped to the [0,1] range according to Equation (1), where Max is the maximum value of each feature and Min is the minimum value of each feature.

3. RFE-based feature extraction method for intrusion detection model.

3.1. Recursive de-featured.

Based on the encoding and normalization of the dataset in the previous section, there is also data redundancy and noise in the dataset itself, and there will be large errors in the actual model training process, which will affect the accuracy of the model.

Recursive feature elimination results in the optimal combination of variables that maximize model performance by adding or removing specific feature variables. The decision tree benchmark model is used for multiple rounds of training, and after each round of training, a feature with several weight coefficients is trained in the next round based on a new feature set. For predictive models with features with weights, RFE selects features by recursively reducing the size of the feature set examined, firstly, the predictive model is trained on the original features, and each feature is assigned a weight. After that, the

features with the smallest absolute weights are kicked out of the feature set. And so on until the number of features remaining reaches the desired number of features.

3.2. **RFE feature extraction model.** The purpose of constructing a recursive feature extraction model is to reduce the dimension, eliminate the redundancy between features, and select the optimal feature subset in the intrusion detection data set.

The algorithm pseudocode is shown below:

RFE feature extraction algorithm
Enter the initial feature set
Iterative loops on the data
Divide the data into training and test sets
The benchmark model is used to train and adjust the model in the training set
Validation testing
Calculate the importance or ranking of each subset
Loop each subset
Extract the top K important variables
The model is trained and tuned in the training set
Validation testing
Calculate the importance or ranking of each subset
End
End

4. **The gray wolf optimization model determines the model hyperparameters.** The selection of some hyperparameters in the constructed model can often make the model have better accuracy, and the inappropriate selection of hyperparameters will lead to the problem of underfitting or overfitting, which also exists in the model constructed in this paper, so the model of gray wolf optimization algorithm is constructed to find the optimal model parameters, so that the model training has better results.

Grey Wolf Optimization Algorithm (GWO), inspired by the grey Wolf. GWO algorithm simulates the leadership hierarchy and hunting mechanism of natural gray wolves. Four types of gray wolves, such as, α, β, δ, ω, and, were used to simulate leadership. In addition, the three main steps of hunting were realized: finding prey, encircling prey, and attacking prey. The GWO algorithm mathematically models the social hierarchy of the gray Wolf, taking the optimal solution as α the second and third best solutions are named and, respectively, β and δ the remaining candidate solutions are assumed to be ω.

The act of a gray wolf rounding up its prey is defined as:

$$D = |C \cdot X_P(t) - X(t)| \tag{2}$$

$$X(t+1) = X_P(t) - A \cdot D \tag{3}$$

Equation (2) represents the distance between the individual and the prey, and Equation (3) is the position update Equation of the gray Wolf. Where t is the current iteration algebra, X_p and X is the coefficient vector, and is the prey position vector and the gray Wolf position vector respectively. A and C are calculated as follows:

$$A = 2a \cdot r_1 - a \tag{4}$$

$$C = 2 \cdot r_2 \tag{5}$$

Where a is the convergence factor, and as the number of iterations decreases linearly from 2 to 0, r_1 and r_2 take random numbers between [0,1].

Gray wolves can identify the location of their prey and surround them. When the gray Wolf recognizes the location of the prey, β and δ, led by α, guide the pack to surround the prey. The mathematical model for individual gray wolves to track prey locations is described as follows:

$$\begin{aligned} D_\alpha &= |C_1 \cdot X_\alpha - X| \\ D_\beta &= |C_2 \cdot X_\beta - X| \\ D_\delta &= |C_3 \cdot X_\delta - X| \end{aligned} \tag{6}$$

Where D_α, D_β, D_δ represent the distance between α, β and δ and other individuals, respectively.

X_α, X_β and X_δ represent the current position of α, β and δ respectively; C_1, C_2, C_3 are random vectors that X are the current location of the gray Wolf.

$$\begin{aligned} X_1 &= X_\alpha - A_1 \cdot (D_\alpha) \\ X_2 &= X_\beta - A_1 \cdot (D_\beta) \\ X_3 &= X_\delta - A_1 \cdot (D_\delta) \end{aligned} \tag{7}$$

$$X\,(t+1) = \frac{X_1 + X_2 + X_3}{3} \tag{8}$$

Equation (7) defines the step size and direction of individual ω and X_α, X_β and X_δ in the Wolf pack respectively, and Equation (8) defines the final position of.

When the prey stops moving, the Wolf completes the hunt by attacking. In order to simulate approaching prey, when the value of a decreases linearly from 2 to 0 during iteration, its corresponding value of A also changes within the interval $[-a, a]$. When the value of the Wolf is within the interval, the next location of the Wolf can be anywhere between its current location and the location of the prey. When $|A| < 1$, the wolves attack their prey (falling into local optimality). When $|A| > 1$, the gray Wolf is separated from the prey, hoping to find a more suitable prey (global optimal). The grey Wolf optimization algorithm model was constructed, and the number of RFE features selected, the number of deep forest cascade layers used in subsequent model construction, the number of estimators for each cascade layer, and the number of trees in each estimator were taken as the dimension vector of the grey Wolf optimization algorithm. The optimization flow chart is shown in Figure 1.

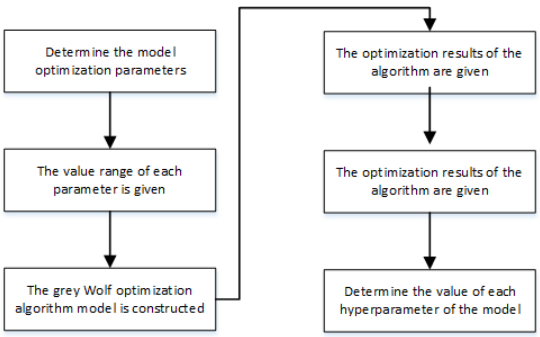


FIGURE 1. Flow Chart of Grey Wolf Optimization Algorithm

5. **Network intrusion detection model.** The network intrusion detection model is a multi-classification detection model, which can not only determine whether an attack has occurred, but also determine which attack type of attack data is available during data analysis.

In this paper, an intrusion detection model based on deep learning is constructed to compare the advantages of applying the proposed GWO-DF intrusion detection algorithm to network intrusion detection.

5.1. **Network intrusion detection model based on deep learning.** Build a deep learning network intrusion detection model, which consists of a layer of CNN and Bi-LSTM, and a layer of Reshape and Batch Normalization. This is shown in Table 2.

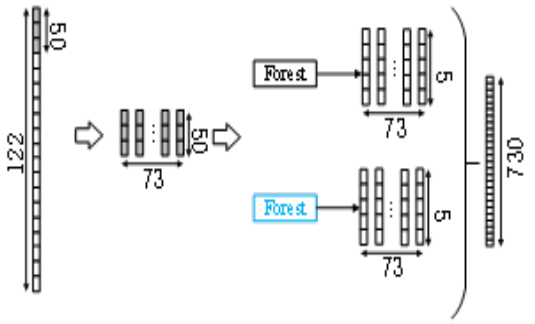
The one-dimensional CNN layer and the maximum pooling layer are used to realize the characteristics of parameter sharing, spatial arrangement and local awareness. Parameter sharing allows for a reduced set of parameters and free variables, allowing features to be extracted with less processing resource usage. Spatial arrangement allows for arrangement in a sparse matrix of features identified so far to better identify correlations between features. Finally, local awareness allows for a reduction in the number of parameters and therefore a significant reduction in the duration of training. Thus, one-dimensional CNNs allow fast-paced spatial learning of a given time series data. The 1-D CNN layer is followed by the Max Pooling layer, which allows sample-based parameter discretization to identify relevant features, reducing training time and preventing overfitting. After Max Pooling, there is the Batch Normalization layer, which normalizes parameters between middle layers to prevent training times from slowing down. The Bi-LSTM layer is used to learn from forward and backward time series data, and the hidden layer uses two cells with the same input and connected to the same output. One unit deals with the forward time series and the other with the reverse time series. This so-called arrangement is said to provide future data for the layers to increase training time and better learn features, thus providing greater accuracy for long-span time series data. The two Bi-LSTM layers in the model are arranged in such a way that each iteration doubles its core size. According to the block diagram of the model, the first Bi-LSTM layer starts with 64 cells, and the next and final Bi-LSTM layer starts with 128 cells. The reason for this choice is to mimic the use of coarse-grained to fine-grained learning to better understand the correlation of long-term time-dependent features of the first 1-DCNN stratigraphic learning, which provides better feature extraction and faster training times. Between each Bi-LSTM layer, there is a MaxPooling layer to eliminate the least relevant features, while a BatchNormalization layer normalizes the output data from the previous intermediate layer to improve performance and reduce training time.

5.2. **Intrusion detection model based on GMO-DF.** Deep neural forests inherit the advantages of neural networks while avoiding the defects of neural networks. Better results were achieved by replacing the layers in the neural network with multiple random forests. The architecture is shown in Figure 2.

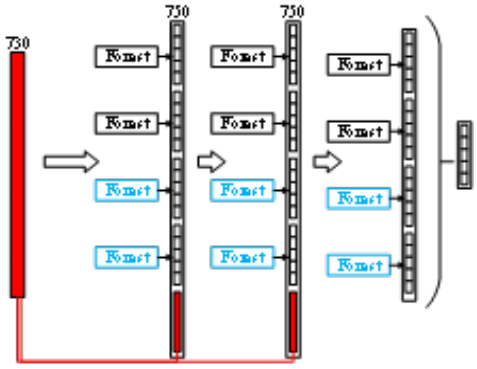
Figure 2 (a) shows the multi-granularity scanning process of the deep forest, the sliding window is selected as 50, the step size is 1, and the specific scanning process is similar to the movement of the convolutional kernel in the convolutional neural network, which mainly amplifies the dimension of the data, and finally cuts it into the data result of size [73,50], and constructs 730-dimensional feature data as input in the cascaded forest after two random forest processing. Figure 2 (b) shows the composition of the cascaded forest in the deep forest, where each layer of the cascade forest consists of two completely random forests (black) and two random forests (blue). The main difference between the two types of forests is the candidate feature space, the complete random forest is to randomly select

TABLE 2. Parameters of CNN based intrusion detection model

layer	output	parameter
convld	(None,122,64)	7872
Max_polling1D	(None,24,64)	0
batch_normalization	(None,24,64)	256
bidirectional	(None,128)	66048
reshape	(None,128,1)	0
Max_polling1D	(None,25,1)	0
batchnormalization	(None,25,1)	4
bidirectional1	(None,256)	133120
dropout	(None,256)	0
dense	(None,5)	1285
activation	(None,5)	0



(a)



(b)

FIGURE 2. Deep Forest Framework

features in the complete feature space to split, while the ordinary random forest is to select the split nodes through the Gini coefficient in a random feature subspace. The input features are input to train two completely random tree forests and two random forests, and each forest is trained independently of the supervised learning. The new output result is merged with the original input features to form a new input, which is generated at the next level. After each layer of the forest is trained, the performance of the model in that layer is tested to see if the performance of the model in the test set continues to improve. If the improvement is not significant, no new cascade forests will be generated.

For the constructed DF intrusion detection model, the RFE feature extraction model proposed in the previous chapter is used to optimize the input feature vectors of the model, and the gray wolf optimization algorithm is used to optimize the hyperparameters used in the model, and the final accuracy of the model is taken as the fitness function of the gray wolf optimization algorithm, the optimization results of the gray wolf optimization algorithm are shown in Figure 3, and the feature selection results are shown in Table 3. In Figure 3, the horizontal axis represents the number of iterations, and the number of

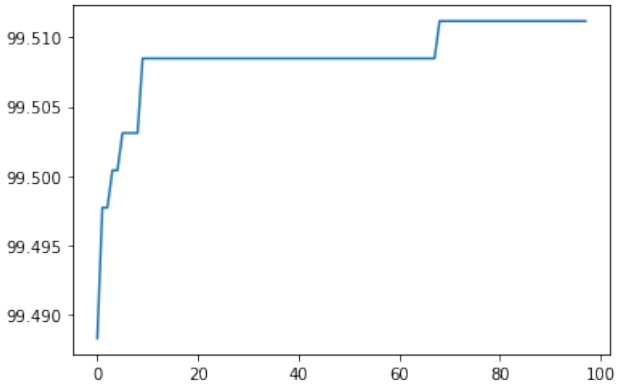


FIGURE 3. Flow Chart of Grey Wolf Optimization Algorithm

iterations of the model built in this paper is selected as 100 times, and in the gray wolf optimization process, the method of cross-validation is adopted, and each round of gray wolf optimization process is processed in a cross-certified way, so when the gray wolf starts iteration, the accuracy is relatively good. Subsequently, the optimal feature subset vector was extracted through feature selection engineering, and more lightweight model results were obtained.

TABLE 3. Parameters of CNN based intrusion detection model

Dataset	Characteristic dimension	Optimal feature	output
NSL-KDD	122	122	99.503
NSL-KDD	122	50	99.511

Table 3 shows the optimal feature subset vector found by the GWO algorithm, which contains 50 dimensional features, compared with the 122-dimensional features, the fitness value is relatively small, but the dimension of the features is reduced by more than one-half, which makes the model more convenient and faster in the construction and training process.

The number of cascading layers in the deep forest used in the final model construction is 20, the estimators for each cascade layer are 2, the number of trees in each estimator is 100, and the threshold for stopping training is set to 1e-5.

6. **Experimental simulation and analysis.** The experimental environment used in this paper is shown in Table 4 below.

The experiments use the datasets mentioned above, and the algorithms are based on the perceptual learning algorithm, the logistic regression algorithm, the neural network algorithm, and the two learning models mentioned above for comparison.

Table 4. Introduction to Experimental Environment

Type	Parament
Operation system	Windows 10
processor	AMD R7-4800H
Number of cores/threads	8 cores /16 threads
Internal memory	16G
Graphics card	RTX 2060
Python	3.8
Development tool	Jupyter notebook

6.1. **Performance evaluation index of the model.** In order to evaluate the performance of the model, the main evaluation indicators used in this paper are Accuracy (Acc), Recall (Re), False Alarm (FA) and Precision (Pre). And by analyzing the confusion matrix of the results, the confusion matrix is the simplest, most basic and most direct way to measure the learning ability of the model. The confusion matrix is shown in Table 5.

Table 5. Confusion matrix

	Predicted value =0	Predicted value =1
True value =0	TP	FN
True value =1	FP	TN

Among them, TP is actually a positive sample, and it is predicted to be a positive sample.

FN is actually a positive sample and predicted to be a negative sample.

FP is actually a negative sample, and the prediction is a positive sample.

TN is actually a negative sample and predicted to be a negative sample. Accuracy is expressed as the proportion of a sample that is correctly predicted to a total sample.

$$Acc = \frac{TP + TN}{TP + FP + TN + FN} \tag{9}$$

Recall: Recall represents the proportion of a sample that is correctly predicted out of the total positive sample. The Equation is shown in (10).

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

The false positive rate indicates the proportion of a negative sample predicted to be positive out of the total negative sample. The Equation is shown in (11).

$$FA = \frac{FP}{FP + TN} \tag{11}$$

Precision indicates the proportion of positive samples that are correctly predicted to the total number of samples predicted as positive, and its Equation is shown in (12).

$$Pre = \frac{TP}{TP + FP} \tag{12}$$

6.2. **Experimental simulation.** The preprocessed data were integrated and divided into training sets and datasets, and the perceptual learning, logistic regression, neural network, deep learning and deep forest algorithms were used for training, respectively, and the training results of each training model are shown in Figures 4-8 below.

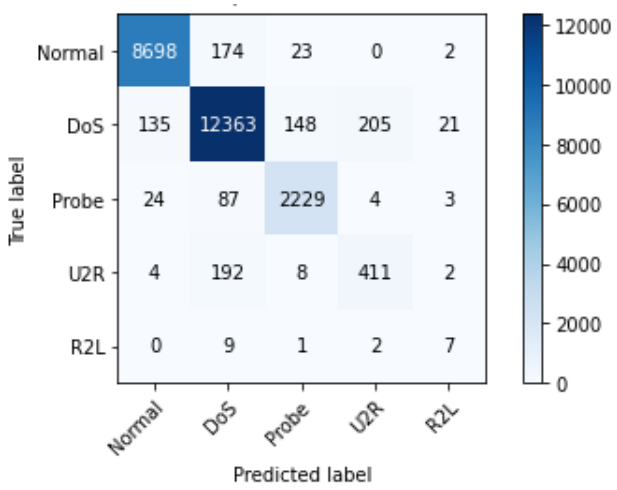


FIGURE 4. **Perceptual Learning Training Results**

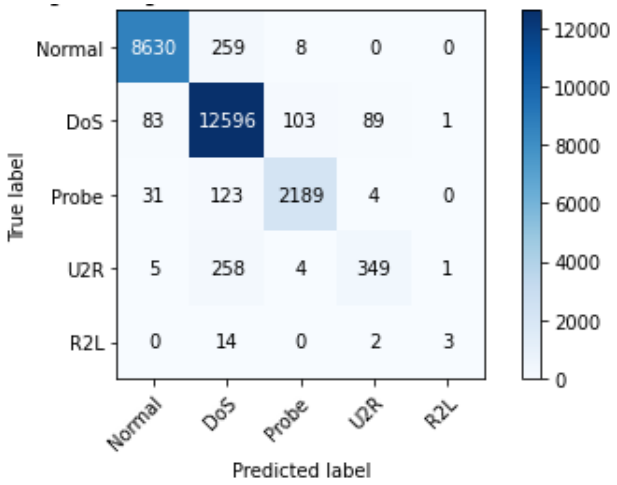


FIGURE 5. **Logistic regression Training Results**

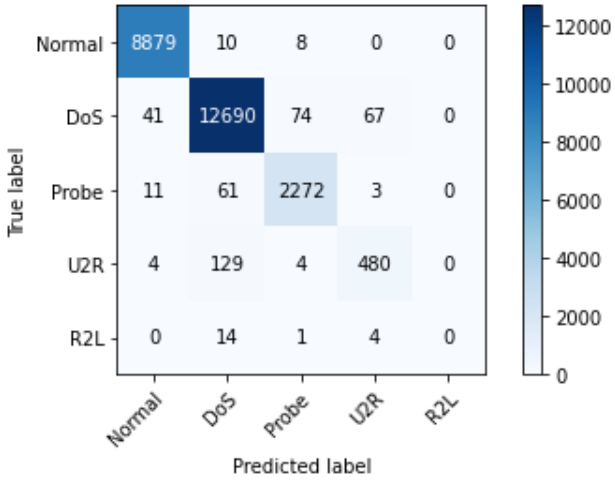


FIGURE 6. **Neural Network Training Results**

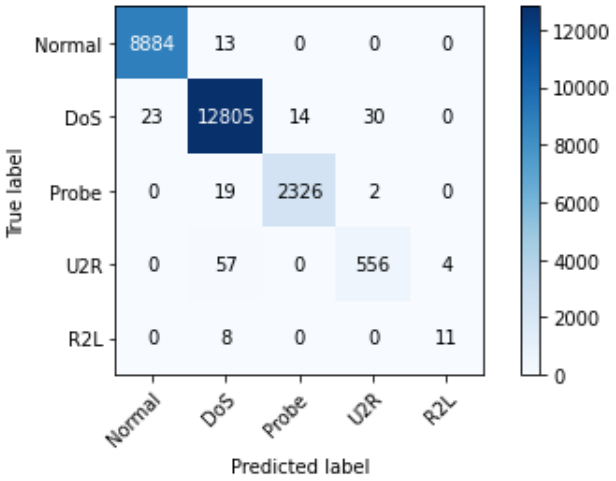


FIGURE 7. **Deep learning training results**

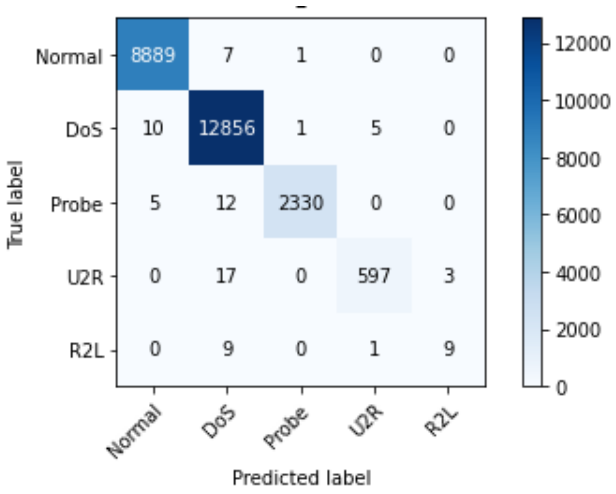


FIGURE 8. **Deep Forest Training Results**

According to the training results of each model, the evaluation indicators described above were used for evaluation. The evaluation pairs of each model are shown in Figure 8. The results of the evaluation are shown in Table 6.

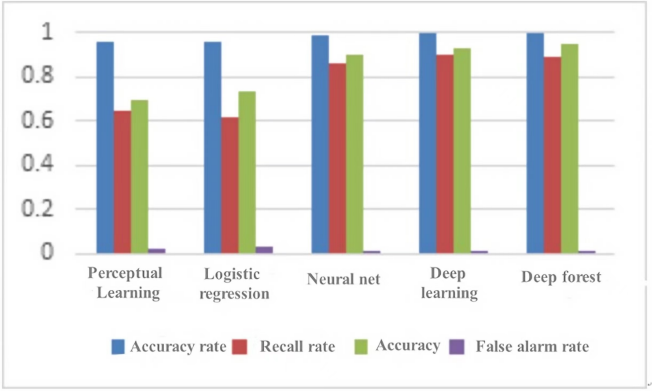


FIGURE 9. **Comparison of Model Evaluation Indicators**

Table 6. Evaluation Indicators of Each Model

Type	Accuracy	Recall	FA	precision
Perceptual learning	0.957	0.644	0.023	0.694
Logistic regression	0.960	0.620	0.031	0.735
Neural network	0.989	0.857	0.003	0.897
Deep learning	0.993	0.902	0.002	0.930
GWODF	0.995	0.886	0.001	0.946

7. **Conclusions.** The comparison results show that the intrusion detection model based on deep forest is 4% higher than the perceptual learning model, 3.7% higher than the logistic regression model, 0.8% higher than the neural network model, and 0.2% higher than the intrusion detection model based on deep learning. Moreover, the model is reduced by more than half in dimension, and compared with other models, the construction and training of the model are lighter and faster, and the comparison shows that the deep forest has the best performance in the accuracy of each model.

REFERENCES

[1] J.-L. Li, H. Zhang, "Survey on semi-supervised anomaly traffic detection," *Journal of Chinese Computer Systems*, vol. 41, no. 11, pp. 2371-2379, 2020.

[2] B.-S. Bhati, C.-S. Rai, "Analysis of support vector machine-based intrusion detection techniques," *Arabian Journal for Science and Engineering*, vol. 45, no. 11, pp. 2371-2383, 2020.

[3] A. Befekadu, "Enhancing the performance of network intrusion detection system by combining naïve bayes, decision tree and k-nearest neighbors Algorithms," *International Journal of Computer Applications*, vol. 180, no. 49, pp. 48-53, 2018.

[4] J.-N. Wang, B. Zhu, W.-X. Yu, "Side channel analysis attack based on deep learning LSTM," *Computer Engineering*, vol. 47, no. 10, pp. 140-146, 2021.

[5] W. Liang, K.-C. Li, J. Long, "An industrial network intrusion detection algorithm based on multi-feature data clustering optimization model," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 2063-2071, 2020.

[6] K. He, C. Qu, X.-J. Zong, "Research and application of machine learning in industrial network intrusion detection," *Journal of Chinese Computer Systems*, vol. 42, no. 2, pp. 437-442, 2021.

[7] p.-p. Kundu, L. Anatharaman, T. Truong-Huu, "An empirical evaluation of automated machine learning techniques for malware detection," in *Proceedings of the 2021 ACM Workshop on Security and Privacy Analytics*. ACM, 2021, pp. 75-81.

[8] S.-C. Zhang, X.-Y. Xie, Y. Xu, "Intrusion detection method based on a deep convolutional neural network," *Journal of Tsinghua University (Science and Technology)*, vol. 59, no. 1, pp. 44-52, 2019.

[9] J. Sun, X. Wang, N. Xiong, "Learning Sparse Representation with Variational Auto-Encoder for Anomaly Detection," *IEEE Access*, vol. 6, pp. 33353-33361, 2018.

[10] P. Wei, Y. Li, Z. Zhang, "An optimization method for intrusion detection classification model based on deep belief network," *IEEE Access*, vol. 7, pp. 87593-87605, 2019.

[11] A. Wlsaeidy, S. Munasinghe, D. Sharma, "Intrusion detection in smart cities using Restricted Boltzmann Machines," *Journal of Network & Computer Applications*, vol. 135, pp. 76-83, 2019.

[12] T. Kim, S.-C. Suh, H. Kim, "An encoding technique for CNN-based network anomaly detection," in *2018 IEEE International Conference on Big Data (Big Data)*. IEEE, 2018, pp. 2960-2965.

[13] T.-T.-H. Le, J. Kim, H. Kim, "An Effective Intrusion Detection Classifier Using Long Short-Term Memory with Gradient Descent Optimization," in *International Conference on Platform Technology & Service*. IEEE, 2017, pp. 1-6.

[14] M.-E. Aminanto, R. Choi, H.-C. Tanuwidjaja, "Deep abstraction and weighted feature selection for Wi-Fi impersonation detection," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 3, pp. 621-636, 2018.

[15] X. Guo, X. Li, R. Jing, "Intrusion Detection Based on Improved Sparse Denoising Autoencoder," *Journal of Computer Applications*, vol. 39, no. 3, pp. 769-773, 2019.

[16] Y.-F. Liu, S. Cai, H.-X. Yang, "Network intrusion detection method integrating CNN and BiLSTM," *Computer Engineering*, vol. 45, no. 12, pp. 7, 2019.

[17] A. Terai, T. Chiba, "Intrusion detection using long short-term memory model for industrial control system," *International Journal of Safety and Security Engineering*, vol. 10, no. 2, pp. 183-189, 2020.
[18] D. Zhang, Z. Yue, X.-X. Zang, "An abnormal intrusion detection method of surveillance video based on self-organizing mathematical model," *Journal of Northeast Electric Power University*, vol. 42, no. 4, pp. 63-69, 2022.
[19] C.-Y. Shao, H.-G. Dou, H.-D. Zhang, "Recursive variational modal decomposition for noise reduction of power transformer sound signals," *Journal of Northeast Electric Power University*, vol. 42, no. 5, pp. 90-96, 2022.
[20] J.-W. Bao, Q.-S. Bao, "Quantitative analysis on the proportion of renewable energy generation based on improved empirical wavelet energy entropy," *Journal of Northeast Electric Power University*, vol. 42, no. 5, pp. 33-43, 2022.