Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105
Volume **9**, Number **3**, August **2024**

Network Malicious Data Intrusion Detection Combining Distributed Network and Improved RF Algorithm under Spark Framework

Jing Zhang*

School of Software and Big Data
Changzhou College of Information Technology, Changzhou 213164, China
zhangjing@ccit.js.cn

Dong-Min Zhao

University of Perpetual Help System Laguna
City of Biñan, Laguna, Philippines 4024, Philippines
z409871218@163.com

*Corresponding author: Jing Zhang
Received December 1, 2023, revised March 17, 2024, accepted June 1, 2024.

Abstract. *With the continuous improvement of network infrastructure, network attacks are more complex. Meanwhile, the continuous expansion of the network scale, increasingly complex structure and endless hacker intrusion methods pose enormous challenges to network intrusion detection. The current intrusion detection system is difficult to face the large-scale high-speed network environment detection. Given the high complexity of network data, this study improves the Random Forest algorithm to construct an intrusion detection model. Then an intrusion detection system is implemented under the Spark framework. The experimental results show that the F1 values of various labels trained by the random forest algorithm are 0.964, 0.963, 0.774, 0.778, 0.762, 0.953, and 0.871. The macroF1 value of the algorithm is 0.867. The F1 values of various labels trained by the improved random forest algorithm are 0.984, 0.982, and 0.875, respectively. The macroF1 value of the improved forest algorithm is 0.907. Moreover, the model training and classification time of the improved random forest algorithm is much shorter, which is improved by about 25%. In conclusion, this designed improved random forest intrusion detection system has higher accuracy and stronger performance. It improves the security of network usage and maximizes the security of user networks.*
Keywords: internet; network security; random forest algorithm; intrusion detection; spark

1. **Introduction.** With the rapid development of Internet technology, network security has become an important issue in modern society [1]. The popularization of network infrastructure not only facilitates information exchange, but also brings new security challenges, especially for malicious data intrusion on the network [2]. These intrusions not only pose a threat to individual users, but also have serious impacts on national security and business operations. In this context, the development of effective network intrusion detection systems (IDS) is particularly important [3]. This research aims to develop an efficient network intrusion detection system for malicious data in the network by combining distributed network and improved random forest (RF) algorithm under the Spark framework. The goal of this system is to achieve highly accurate malicious data

detection in large-scale and high-speed network environments. Considering the limitations of existing intrusion detection methods, especially the efficiency when dealing with large data, this research focuses on the optimization and implementation of the algorithm, which is committed to improving the detection speed and accuracy. In existing research, there are some disagreements about the methods and effectiveness of network intrusion detection. Especially in terms of algorithm selection and optimization, different researches have proposed various views. In this study, based on these disagreements, an improved RF algorithm is proposed, implemented and tested in the Spark framework. It aims to explore more effective cybersecurity solutions. Taken together, the main contribution of this research is to develop an efficient intrusion detection system for large-scale network environments. By combining the improved algorithm with the Spark framework, this research not only improves the accuracy of intrusion detection, but also provides an effective solution for handling large amounts of data. The experimental results show that the proposed method has a wide range of applications in the field of network security, providing new perspectives and technical support for related research.

2. **Related Work.** With the development of the Internet, network attacks are increasingly frequent and the IDS is proposed to protect the data security of users to some extent. Althobaiti et al. created a IDS based on cognitive computing to achieve security in the physical systems of industrial networks. The system covered data collecting, pre-processing, feature selection, classification, and parameter optimization. The noise present in the collected data was removed. The model was optimized to select the optimal subset of features for final intrusion detection. The outcomes indicated that the proposed model had good performance [4]. According to Devi et al.'s research, the information technology infrastructure and traditional operating systems used in cloud computing are susceptible to intrusion attacks. To address the cloud computing security problem, the research team proposed an adversarial network-breeding IDS with dual-channel capsule generation optimized by the Red Fox optimization algorithm. The research results showed that the system had a large improvement in accuracy and performance than the traditional method [5]. Zhang et al. established a IDS based on the defense-in-depth concept to enhance the network security of industrial control systems. The system had multiple layers of defense, providing time for the compromised system. The research team simulated five attacks, including false data injection, data filtering, man-in-the-middle, data tampering, and denying services, to provide a second layer defense in the event that the intrusion detection defense layer fails. The findings revealed that the system can detect network intrusions before major consequences occur, effectively providing time for the system [6]. Ning et al. analyzed the controller local area network. Due to the lack of security protection mechanisms in the controller area network, various attacks on the controller area network posed a serious threat to the safety of vehicles. To solve this problem, the research team proposed a IDS based on local anomaly factor, utilizing the characteristics of voltage signals on the controller LAN bus. The findings suggested that the method could significantly increase detection precision, avoid the modification of the controller LAN protocol and reduce the computational content [7]. Detecting nodes that propagate false data is a prerequisite for effectively deploying connected vehicle network services. Anyanwu et al. proposed a novel overshooting ensemble into RF algorithm for detecting false underlying security messages in connected vehicle networks. The results showed that the proposed algorithm far outperformed other algorithms with 99.60% [8]. With the rapid growth of wind energy production and manufacturing, the cost of operating and maintaining the engines is also increasing rapidly. Most of the wind turbines are equipped with supervisory control and data acquisition systems for system control and

data logging. Qin et al. proposed using various supervisory data as data points. The wavelet analysis was applied to reduce the noise in the input signals. Finally, the recursive least square filter was applied to reduce the false alarm rate and find the optimal input features. The results showed that the RF algorithm used provided more accurate output and significantly reduced the false alarm rate [9]. To mitigate network attacks, Haffar et al. explained the false classification rate of the federated model by building a decision tree RF model. Firstly, a RF containing depth constrained decision trees was used as an alternative to the federated black box model. Then the decision trees in the forest were used to calculate the feature values in the erroneous prediction model. The results showed that the model was capable of detecting malicious attacks on the network with high accuracy [10].

In summary, in the research on dealing with malicious network intrusion, more researchers realize that the traditional network intrusion system cannot fully meet the current needs. There are problems such as poor accuracy, difficulty in dealing with complex environments, and long iteration times. To address these problems, this research introduces the Spark framework to cope with the many iterations and long time. The RF algorithm is introduced to address the poor detection accuracy in complex environments. Then the random algorithm is improved to optimize the system performance.

3. Network IDS Construction based on Spark Framework and Enhanced RF Algorithm.

With the continuous development of the Internet, 5G technology and Internet of Things (IoT) technology have made great progress, which makes the current network environment increasingly complex. To protect the user demands for system network security, relevant scholars have constructed the IDS model. The traditional IDS is slightly insufficient to face the current complex network environment. Therefore, the research combines the RF algorithm with IDS to develop an improved RF algorithm based on the Spark framework, which is used to effectively detect network malicious data intrusion. To achieve this goal, the research adopts a series of innovative methods and technical strategies. Firstly, the focus is placed on optimizing the RF algorithm to enhance the performance of the model by adjusting the decision tree generation and feature selection mechanisms. Secondly, the distributed computing capability of the Spark framework is utilized to process large-scale network data, improving the processing speed and efficiency of the system.

3.1. IDS and RF algorithm.

The IDS is a system that monitors and defends against malicious intrusions on a network or computer. IDS can proactively detect attacks and traces intrusion points based on network behavior and traces. In addition, IDS can also analyze the suspicious behavior and determine the intrusion type of the behavior by comparing it with the known pattern library. Then the corresponding response operation is made according to the security policy. The working process of traditional IDS is shown in Figure 1.

The operational flow of a IDS, a security mechanism specifically designed to monitor and defend against malicious activity on a network, is shown in Figure 1. The core process of the IDS begins with continuous monitoring of host status, activity, system logs and the network to collect audit data, which can help to promptly identify signs of potential attacks or anomalous behavior that violates security rules. The system then reviews and analyses this audit data in detail, a stage that is key to the detection process. In the intrusion detection segment, various efficient techniques and methods are often utilized to perform intrusion detection. Once abnormal activities are detected, the IDS will activate its management module to react according to the established security policy. Necessary

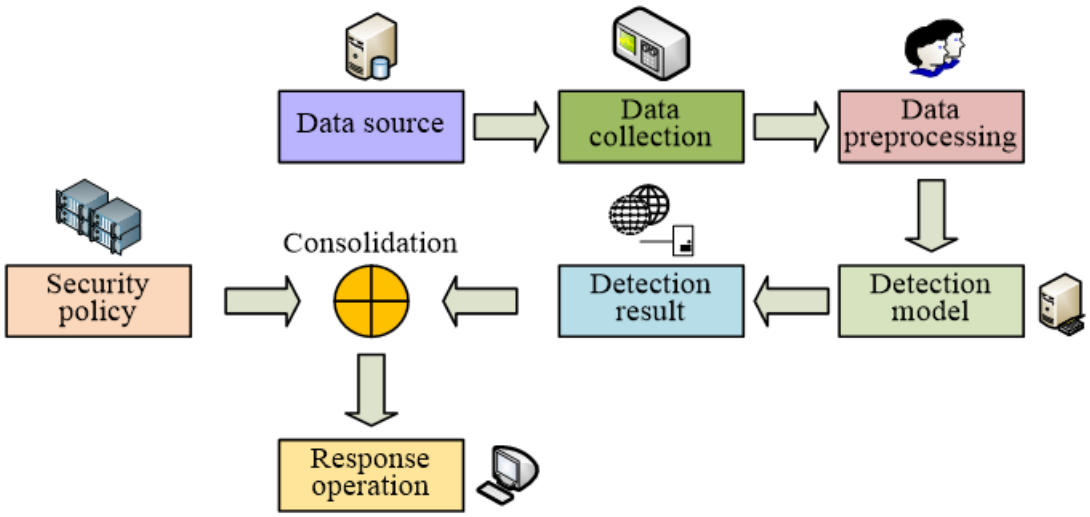

Figure 1. Workflow of intrusion detection system

measures are taken, including logging intrusion details, issuing alerts, and blocking illegal activities.

The RF algorithm is a powerful machine learning method that operates on an unpruned decision tree. Decision tree models are popular in the field of machine learning because they are not only efficient and easy to implement. In addition, their unique processing does not require normalization or standardization of feature values before model training. The decision tree model predicts through a top-down approach. According to a certain decision-making logic, data is classified to ultimately obtain classification results or predicted values. In the RF algorithm, multiple such decision trees are constructed in parallel and each tree is trained on a random subset of the dataset, which helps to increase the diversity and robustness of the model. Each tree works independently in the decision-making process. The final prediction is obtained by integrating the predictions of all trees, usually using a majority vote. This integrated approach effectively reduces the bias and variance of the model and improves the accuracy and reliability of the overall prediction. This integrated learning approach of the RF algorithm combines multiple decision trees, which can effectively handle various complex datasets. It has a wide range of application scenarios. The general structure of a decision tree is shown in Figure 2.

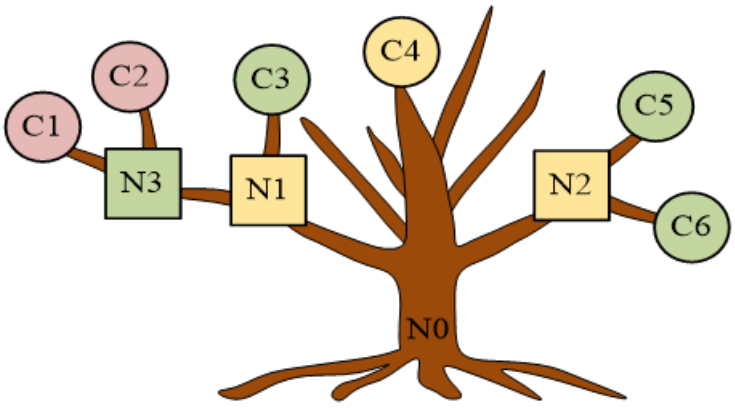

Figure 2. Decision tree structure

Figure 2 shows the decision tree structure. Each node of the decision tree represents a test of a feature or attribute. Directed edges between nodes indicate different results of the test. The top node is called the root node, which is the starting point of the decision-making process. From the root node, the tree diverges along different paths depending on the feature values of the data until it reaches the leaf nodes, which are at the bottom of the decision tree. Each of these nodes corresponds to a classification decision or prediction output. A key advantage of this structure is its intuitive and explanatory nature, making the decision-making process easy to understand and interpret. The node $N_i, (i = 0, 1, 2, 3)$ is the decision node. The node $C_j, (j = 1, \ldots, 6)$ indicates the category of classification. The DT algorithm is a method for creating branching sub-trees and choosing the best features for classification, and then classifying the data through these branches. The DT is constructed in two steps, feature selection and DT construction. How to select the best segmentation features is the key to the DT construction process. In DT, the methods commonly used to measure the segmentation effect of features on samples are information gain, information gain ratio, and Gini index.

The information gain (IG) method uses information entropy to calculate the information gain of each attribute, and determines the best node splitting feature based on the IG. For the discrete source a, the self-information formula characterizing the magnitude of the information transmitted by a_i is shown in Equation (1).

$$I(a_i) = -\log_2 P(a_i) \tag{1}$$

In Equation (1), $P(a_i)$ is the probability of the discrete information source a in the k fetches. To measure the overall uncertainty of the source, the information entropy is introduced, as shown in Equation (2).

$$\text{Info}(D) = E(X) = \sum_{i=1}^{m} P(a_i) \log_2 P(a_i) \tag{2}$$

In Equation (2), X is the source, D denotes the dataset, m denotes the classification. A denotes the feature. From Equation (2), for a dataset D, the information entropy is known. If feature A is applied to classify D, the information entropy of the divided data subset is shown in Equation (3).

$$\text{Info}_A(D) = \sum_{j=1}^{v} \left(\frac{|D_j|}{|D|} \times \text{Info}(D_j) \right) \tag{3}$$

In Equation (3), the number of samples in dataset D is $|D|$. The number of categories classified as jth is $|D_j|$. v denotes the number of subsets classified by feature A. The $\text{Info}_A(D)$ is negatively correlated with the classification purity. The information gain of feature A can be obtained from Equation (2) and (3), as shown in Equation (4).

$$\text{Gain}(A) = \text{Info}(D) - \text{Info}_A(D) \tag{4}$$

The strength of classification ability for feature A is characterized by the magnitude of information gain value. The essence of the information gain method is to calculate the IG value of each attribute value and select the largest attribute to classify the subset. However, the information gain method has the biased multi-value features. The information gain ratio method can effectively solve the problem. The IG ratio is shown in Equation (5).

$$\begin{cases} Gain\_ratio(A) = \frac{Gain(A)}{Split\_Info_A(D)} \\ Split\_Info_A(D) = -\sum_{j=1}^{v} [\frac{|D_j|}{|D|} * \log_2(\frac{|D_j|}{|D|})] \end{cases} \tag{5}$$

In Equation (5), D is the data set. A is the feature. v denotes the number of subsets divided by feature A. Split_Info$_A(D)$ is the penalty coefficient, which is negatively correlated with the size of feature A, thus avoiding the biased features with multiple values. However, this method still has problems. If a feature has a small number of values, the corresponding penalty coefficient will be larger, resulting in a larger information gain ratio. Features with fewer values are more likely to be selected.

$$\begin{cases} Gini_{split\_A}(D) = \sum_{j=1}^{v} [\frac{|D_j|}{|D|} * Gini(D_j)] \\ Gini(D) = 1 - \sum_{i=1}^{m} (\frac{|C_i|}{|D|})^2 \end{cases} \tag{6}$$

In Equation (6), D is the dataset, and A is the feature. v denotes the number of subsets divided by feature A. $|C_i|$ denotes the quantity of i categories in dataset D. The Gini index method is used in node splitting. The Gini index of all features is calculated from Equation (6). Then the best split feature is the one with the lowest Gini index.

RF is a multi-classifier composed of several DTs. The classification result is determined by voting method or taking the mean value. RF uses Bagging integration ideas to build models for work, randomly extracting feature subsets and training subsets for generating multiple DT classifiers, respectively. The generated multiple classifiers are combined for prediction. The RF classifier with integrated learning idea of multiple classifiers is constructed using Bagging integration. Figure 3 depicts the construction procedure.

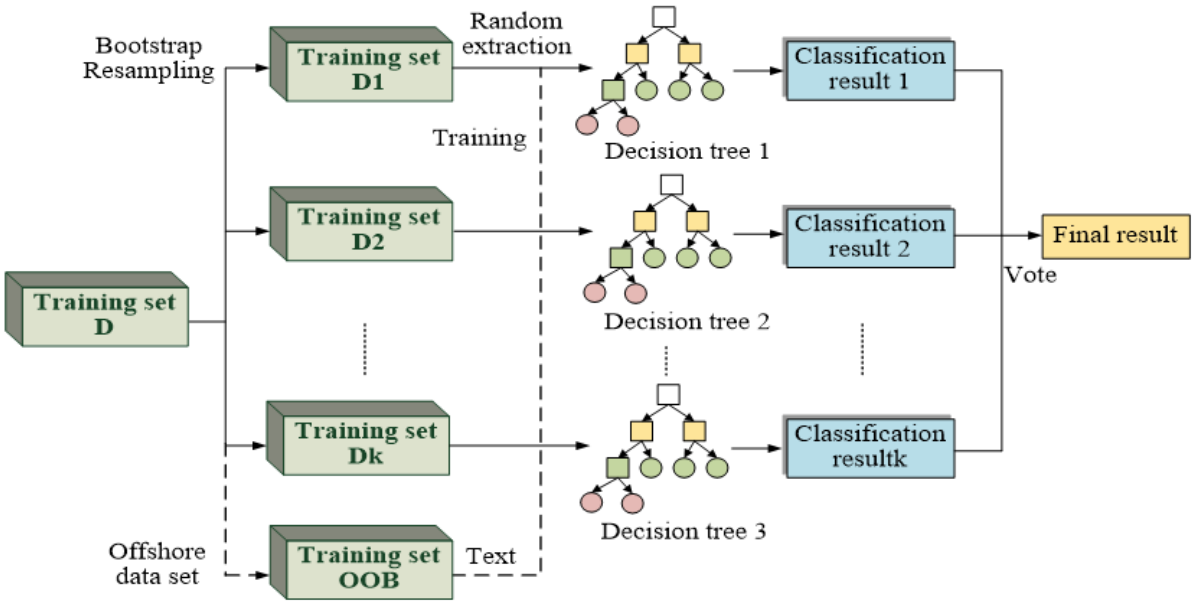

Figure 3. RF construction diagram

From Figure 3, the RF construction is divided into four major components, randomly drawn sample set, randomly drawn split features, DT generation and RF classifier construction. The first part is random sample set extraction. The Bootstrap sampling

method is applied to extract k training sets, $D_{train} = D_1, ..., D_k$. The same samples are taken each time, and no processing is done after taking. The extraction is repeated several times. The samples that are not extracted each time constitute the out-of-bag data set. The second part is to randomly extract segmentation features. m features are extracted randomly and without putting back from M sample features. The extracted features are used as the nodes to generate the DT to divide the attribute set. The third part is DT generation. The common types of DT generation methods are selected. The single DT is trained and constructed based on the above randomly extracted sample set and feature set. Finally, the single DT is trained iteratively to build and obtain multiple DTs. The fourth part is to integrate the individual DTs into a decision forest. In the prediction stage, the sample data are fed into each DT to get the final splitting result [11].

3.2. **Construction of IDS based on improved RF algorithm.** With the current popularity and development of the Internet, the frequency and complexity of malicious cyber-attacks have risen dramatically. This poses unprecedented challenges in the field of network security. The diversity and advanced techniques of modern network attacks, such as Distributed Denial of Service (DDoS) attacks, phishing attacks, and zero-day vulnerability exploits, pose a severe test for traditional security defence mechanisms. In such a context, effective IDSs have become the key to ensure network security. This research is dedicated to constructing an efficient and accurate network intrusion detection model by combining an advanced intrusion detection system framework with an improved RF algorithm. The model aims to improve various network attack detection through innovative methods and techniques, while ensuring processing efficiency in high data traffic environments. The constructed model not only considers the characteristics of existing network attacks, but also foresees new types of attacks that may appear in the future, which makes the system have good adaptability and scalability. The overall construction process of the model is shown in Figure 4. The intrusion detection model in Figure 4

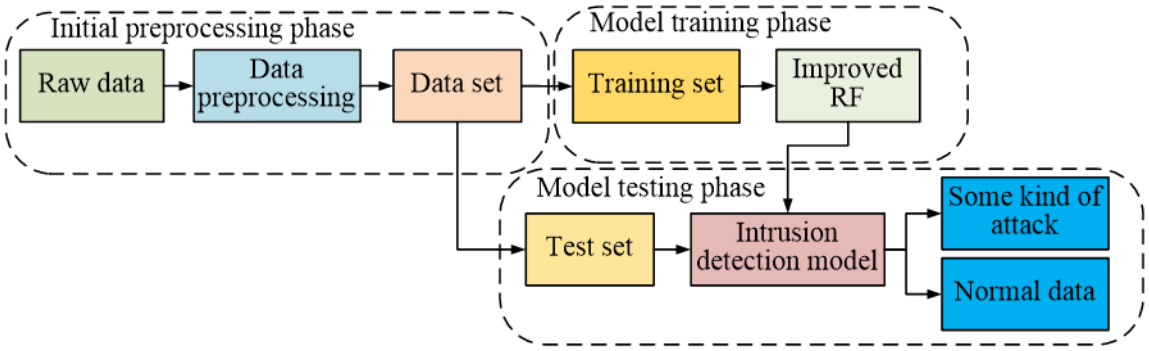

Figure 4. The construction process of IDS

involves several key steps. First, network traffic data is comprehensively collected and preprocessed to ensure the accuracy and integrity of the data. Next, feature selection and data analysis are performed by an improved RF algorithm to effectively distinguish normal traffic from potentially malicious activities. On this basis, the model applies a multi-layer decision-making mechanism to judge and classify network behaviors, thus achieving accurate identification of various network attacks. In addition, the model includes a dynamic learning mechanism that can be continuously optimized and adjusted based on the latest cyber threat data to cope with the ever-changing cyber security threat environment. The whole intrusion detection can be divided into 3 stages. In the data

preprocessing phase, the main preprocessing of network data is performed. In the model training phase the decision tree model for classification detection is generated based on RF. The processed network intrusion dataset is used as the training set for the improved RF algorithm, then multiple weakly classified decision trees with weights are generated. Finally, all the weakly classified decision trees are composed into a multi-classifier model for intrusion detection. In the model testing phase, the test set is classified using the multi-classifier model to determine whether the sample data is attack data and to determine the type of attack. At the same time, the performance metrics of the model are tested using the test set, which can be evaluated in a better way [12].

RF has strong noise resistance, easy to scale, easy to parallelize, and strong generalization ability. However, some shortcomings are more prominent when dealing with large and complex network intrusion. Traditional RF randomly selects feature subsets to build DTs. There are a large number of highly correlated features in network data traffic features, which leads to excessive similarity between DTs. It is not conducive to improving the generalization ability of decision forests. From the nature of RF, as the threshold increases throughout the forest, the generalization error PE converges to an upper bound, as shown in Equation (7) [13].

$$PE \leq \bar{p}(1 - s^2)/s^2 \tag{7}$$

In Equation (7), s is the average classification accuracy of the DT. $\bar{p}$ is the average association between DTs. From Equation (7), increasing s or decreasing $\bar{p}$ can reduce the generalization error PE and improve the model strength. An improved Relief is combined with RF. The enhanced Relief-RF is able to increase the feature expressiveness of a few classes of samples and calculate the feature weights. Then the feature random selection process in generating DTs is restricted by a stratified feature random sampling strategy. It can avoid too many poorly performing features or invalid features from being selected. Each feature subset has better discriminative ability, thereby improving the classification strength of the IDS model [14]. Relief-RF is a feature weight algorithm that measures the discrimination between features and positive and negative categories. Equation (8) displays the weight calculation.

$$\begin{cases} W(A) = \sum\limits_{i=1}^{k} \frac{diff(A,S,N)}{k} - \sum\limits_{i=1}^{k} \frac{diff(A,S,M)}{k} \\ diff(A, S_1, S_2) = \begin{cases} \frac{|S_1[A] - S_2[A]|}{\max(A) - \min(A)}, if A is continues \\ 0, if S_1[A] = S_2[A] \\ 1, if S_1[A] \neq S_2[A] \end{cases} \end{cases} \tag{8}$$

In Equation (8), A denotes the feature, S denotes the sample, and diff(A, S_1, S_2) denotes the distinction between S_1 and S_2 on the feature value A. In Equation (8), a feature A is selected. A sample S is randomly selected along with the neighboring sample M and negative class N. Then the distance between the sample M and negative class N and the feature A is calculated. The weights of the feature A are adjusted according to the difference of the distance between the same and the different. The classification weights of the final feature A is obtained after iterating k [15]. To solve the problem that Relidf algorithm is not applicable to multiple labels, the Relidf algorithm is improved by changing the sample extraction to n nearest-neighbor samples from each category. Then the distance is iteratively updated. The weight is shown in Equation (9).

$$W(\mathrm{A}) = \sum_{i=1}^{k} \frac{\sum\limits_{C \notin c(\mathrm{S})} \left[\frac{p(c(\mathrm{C}))}{1-p(c(\mathrm{S}))} \sum\limits_{j=1}^{n} diff(\mathrm{A},\mathrm{S},\mathrm{N}_j(\mathrm{C})) \right]}{nk} - \sum_{i=1}^{k} \frac{\sum\limits_{j=1}^{n} diff(\mathrm{A},\mathrm{S},\mathrm{M}_j)}{nk} \tag{9}$$

In Equation (9), C is the heterogeneous sample of the sample S. $p(c(C))$ and $p(c(S))$ denote the weight of C and S in the sample set of the respective categories to which they belong, respectively. $N_j(C)$ denotes the sample that is dissimilar to the sample S. Although the improved Relief-RF can effectively solve the shortcomings of the Relief, the randomly selected equal heterogeneous sample is not well suited to the imbalance problem of the intrusion dataset in this study. Therefore, the Relief-RF algorithm is improved again. The improved weight iterative is shown in Equation (10) [16].

$$\begin{cases} W(A) = W(A) - \sum\limits_{i=1}^{k} \frac{\sum\limits_{C \notin c(S)} \left[\frac{p(c(C))}{1-p(c(S))} \sum\limits_{j=1}^{n_2} diff(A,S,N_j(C)) \right]}{n_2} - \sum\limits_{i=1}^{k} \frac{\sum\limits_{j=1}^{n_1} diff(A,S,M_j)}{n_1} \\ n_1 = n * \frac{l_-}{l_+ + l_-}, SisPositive \\ n_1 = n * \frac{l_+}{l_+ + l_-}, SisNegative \end{cases} \tag{10}$$

In Equation (10), n_1 is the like-neighboring sample. n_2 is the dissimilar neighboring sample [17]. The improved Relief-RF algorithm can ensure that the minority class has a larger proportion when positive and negative samples are sampled. It can effectively avoid the problem that the classification results are biased toward the majority class. The improved Relief-RF is used to optimize the RF algorithm random feature sampling subset process to avoid too many poor performance or invalid features selected to generate DTs [18]. The hierarchical random feature selection process is shown in Figure 5.

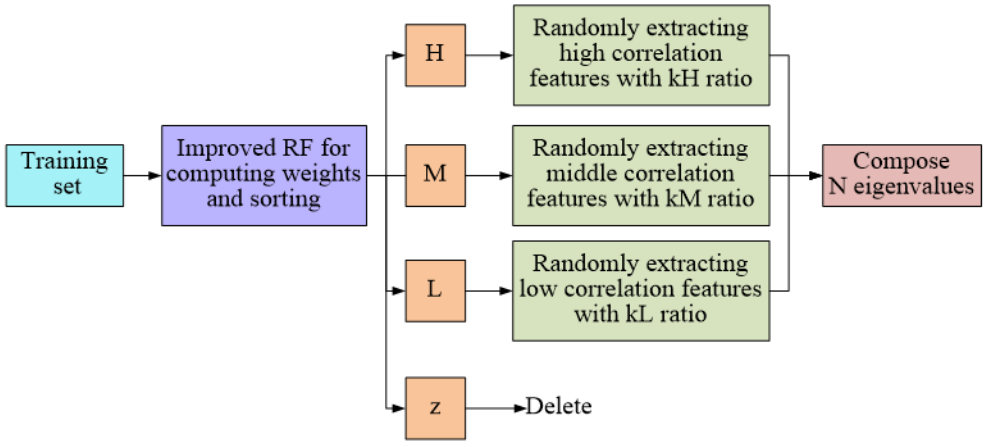

Figure 5. Layered random feature selection process

This experiment improves the Relief-RF algorithm to calculate the weights of network traffic data features and discards zero weight features. The remaining features are arranged into three feature subsets of high, medium and low based on their weights. A DT is generated by extracting features from the three feature subsets of high, medium and low in the ratio of 5:3:2 when constructing the DT. The DT classification accuracy is checked with a test set. It is used as the weight of this DT. The result of the integrated

strategy model is shown in Equation (11) [19].

$$F(X) = \sum_{i=1}^{k} \omega_i \cdot h_i(X_j) \tag{11}$$

In Equation (11), the weight of the decision number $h_i(X_j)$ is ω_i, which is positively correlated with DT classification ability. The IDS model is constructed using the improved RF algorithm. The main process is shown in Figure 6.

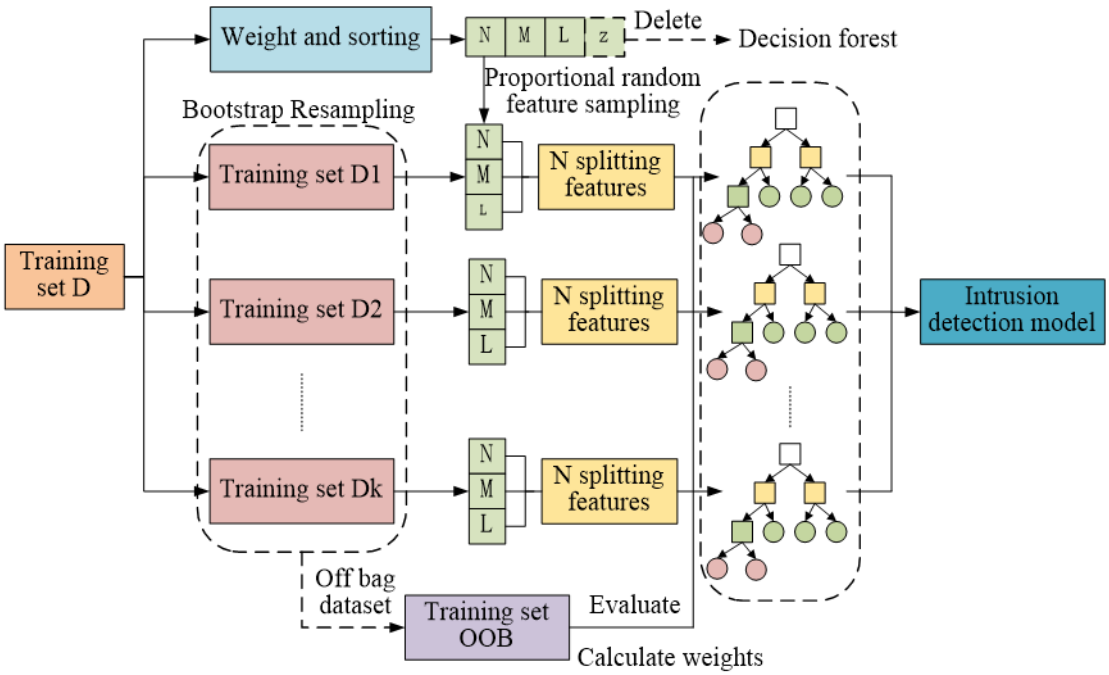

Figure 6. Improved RF build IDS process

Firstly, the weights of each feature are calculated using the improved RF algorithm. Then the features with zero weight are eliminated. The remaining features are arranged according to the weight values and separated into three feature sets, high, medium and low. Then the features are randomly selected from each of the three feature sets in a 5:3:2 ratio in a hierarchical manner for generating DTs. Finally, the accuracy of each DT is evaluated using the out-of-bag dataset. The accuracy is used as the weight of each DT. All DTs are combined with weights to obtain the IDS model [20]. The classification evaluation index is the confusion matrix. Because of the long training time of traditional RF algorithm in processing large amount of network data, Spark distributed framework is introduced to construct the distributed cluster environment and design distributed network IDS. It is divided into network traffic collection module, data transmission processing module, intrusion detection module and intrusion management module.

4. **Base and improved RF algorithm for IDS performance testing.** In this chapter, the network IDS constructed by RF is compared with the network IDS constructed by improved RF. The performance of the two different algorithms for intrusion detection is analyzed. In addition, a new detection algorithm model is introduced for comparison, so as to filter out the model with the best classification performance. The effectiveness of the model proposed in the study is further validated. The experimental results confirm that the improved RF algorithm can show excellent performance on several standard

datasets, especially in the key performance metrics such as classification accuracy, recall and F1 score. Moreover, by comparing with the traditional RF algorithm and other popular algorithms, the results also demonstrate the significant advantages of the improved algorithm in network intrusion detection.

4.1. **Data processing and environment setup.** Most current public datasets lack diversity in traffic characteristics. There are fewer attack types to simulate the most realistic network attack trends. The test dataset is CICIDS2017, a public intrusion dataset published by the North American Institute for Secure Networks. This dataset records network traffic data for five consecutive days, including attacks such as DoS attacks, DDoS attacks, botnets, brute-force attacks, etc. Detailed information is shown in Table 1.

Table 1. CICIDS2017 Data Set

Label	Attack type	Number	Total (numerical)
Benign	Benign	2273097	2271320
	DDoS	128027	128025
DoS/DDoS	DoS Slowloris	5796	5796
	DoS Slowhttptest	5499	5499
	DoS Hulk	231073	230124
	DoS GoldenEye	10293	10293
	Heartbleed	11	11
Brute Force	FTP-Patator	7938	7935
	SSH-Patator	5897	5897
Web Attack	WebAttack-BruteFore	1507	1507
	Web Attack-XSS	652	652
	WebAttack-SqlInjectn	21	21
Infiltration	Infiltration	36	36
PortScan	PortScan	158930	158804
Bot	Bot	1966	1956
Total Intrusion	/	557646	556556
Total samples	/	2830743	2827876

Table 1 displays the detailed data details. From Table 1, this dataset is an unbalanced dataset. The RF algorithm is not effective in classifying the dataset with a small number of samples. Therefore, it is necessary to pre-process this dataset by extracting the different samples in the training set to obtain the training dataset samples, as shown in Table 2.

Table 2. Extract the Number of Samples for Each Category

Type	Benign	DoS/DDoS	Brute Force	Web Attack	Infiltration	PortScan	Bot
Code	0	1	2	3	4	5	6
Original sample	2273097	379748	13832	2180	36	158804	1956
Sample	48297	39978	2076	1349	468	23840	1285

The number of samples for each data category in the training set after pre-processing is shown in Table 2. The processed dataset is used as the training dataset to train the iterations of different models. The detection effect of each model is analyzed.

4.2. **Intrusion detection model performance testing.** To evaluate the effectiveness of various IDSs, the iterative performance of the constructed network intrusion detection models is first tested. The simulation experiments are implemented in Matlab. The network intrusion detection model under the traditional RF algorithm (denoted as RF-IDS), the network intrusion detection model under the improved RF algorithm (denoted as IRF-IDS), the network intrusion detection model under the BP neural network (denoted as BP-IDS) and the network intrusion detection model under the genetic algorithm-optimized BP neural network (denoted as GA-BP-IDS) are compared, respectively.

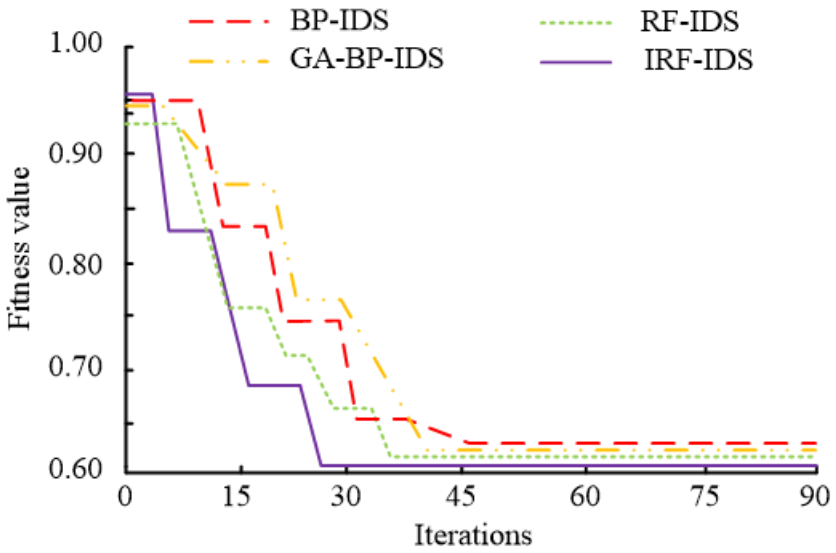

Figure 7. Fitness of different models under different iterations

Figure 7 shows the fitness changes of four models at different iterations. From Figure 7, the IRF-IDS model can reach the stable adaptation value in about 27 iterations, and the best value is 0.61. The RF-IDS model can reach the stable adaptation value in about 36 iterations, and the best adaptation value is 0.62. The GA-BP-IDS model and BP-IDS model can reach the stable adaptation value in about 41 and 46 iterations, respectively. The stable adaptation value is 0.63 and 0.64. Compared with the other three models, the stable adaptation value is 0.63 and 0.64, respectively. Compared with the other three models, the IRF-IDS model can iterate to the stable faster. Therefore, it has better stability in intrusion detection. The IRF-IDS model reaches the stable faster than the other models. This may be due to its optimization in feature selection and classification algorithms, which can process the data more efficiently, thus speeding up the convergence of the model and improving its intrusion detection stability.

The error performance of the four models during the iterative process is shown in Figure 8. The average sum of squares of errors for the four models is shown in Figure 8(a). From Figure 8(a), the average sum of squares of errors values after reaching stable state for IRF-IDS, RF-IDS, GA-BP-IDS, and BP-IDS are 0.24, 0.29, 0.33, and 0.36, respectively. The minimum sum of squares of errors for the four models is shown in Figure 8(b). From Figure 8(b), the minimum error sum of squares values of IRF-IDS, RF-IDS, GA-BP-IDS, and BP-IDS after reaching the stable state are 0.13, 0.17, 0.18, and 0.19, respectively. Based on the error performance of different models, the IRF-IDS model is able to iterate to the stable mean error sum of squares and minimum error sum of squares more quickly. IRF-IDS reaches a converged state, with values of 0.24 and 0.13 for the mean error sum

of squares as well as the minimum error sum of squares, respectively. It indicates that it has higher prediction accuracy and stability, due to the algorithm's advantages in dealing with data noise and complex features.

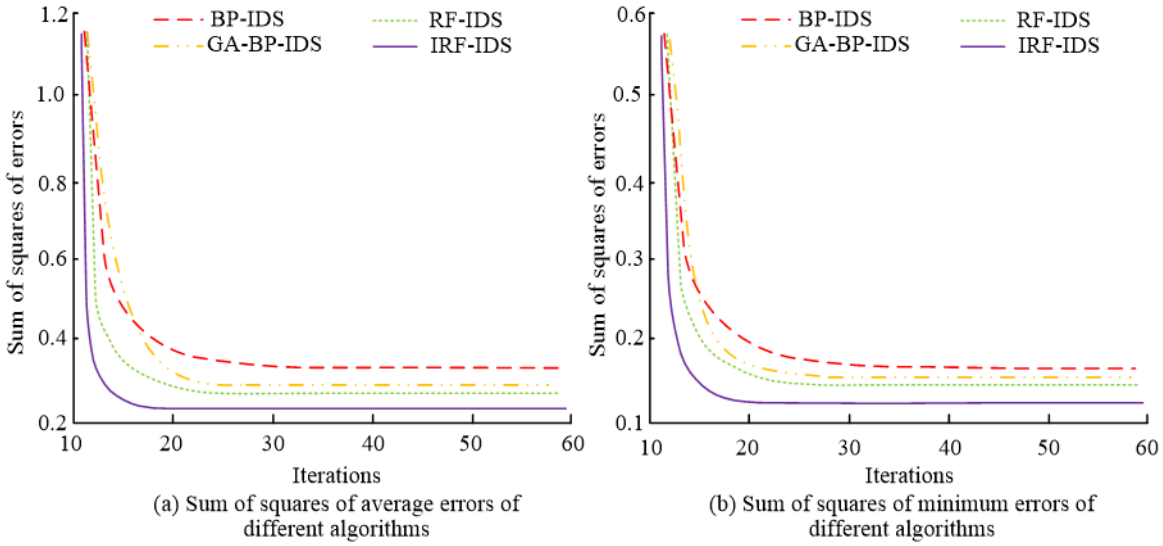

Figure 8. Error performance of different algorithms

The extracted sample dataset in Table 2 is further selected for evaluation tests. The test set is categorized and predicted using the conventional RF algorithm as the reference algorithm. The confusion matrix shown in Table 3 is obtained. The first column is the prediction type of the algorithm and the first row is the attack category label.

Table 3. RF Algorithm Prediction Result Confusion Matrix

PTT	**0**	**1**	**2**	**3**	**4**	**5**	**6**
0	46436	728	203	78	32	463	58
1	773	38488	117	49	23	519	46
2	309	139	1698	17	12	122	17
3	219	118	19	1153	9	79	19
4	57	28	4	3	368	31	7
5	473	410	27	35	17	22573	12
6	30	67	8	14	4	53	1126

Table 3 shows the prediction results of the RF algorithm on different attack types. Each element in the confusion matrix represents the prediction performance of the algorithm on a specific category. Higher values of true positives and lower values of false positives indicate that the algorithm performs well on certain categories. Meanwhile, they also reveal the limitations of the algorithm on certain categories. For example, a high false-negative rate indicates poor detection of certain attack types. From this table, the F1 values of RF algorithm for each type of label in the training set are 0.964, 0.963, 0.774, 0.778, 0.762, 0.953, and 0.871. The macroF1 value of RF algorithm is 0.867. The classification prediction experiment of the improved RF algorithm model is designed. Table 4 displays the test results.

From the data in Table 4, the F1 values for each type of labels in the training set under the improved RF algorithm are 0.984, 0.982, 0.875, 0.842, 0.796, 0.982, and 0.889,

Table 4. Improved RF Algorithm Prediction Result Confusion Matrix

PTT	0	1	2	3	4	5	6
0	47580	319	102	71	31	235	52
1	354	39237	61	39	27	139	37
2	117	103	1861	14	13	56	13
3	92	86	20	1181	8	52	16
4	29	24	3	2	362	19	3
5	97	148	21	27	23	23293	7
6	28	61	8	15	4	46	1157

respectively. The macroF1 value for the improved RF is 0.907. The data in Table 4 shows the F1 values of the improved RF algorithm on different categories. Compared with the traditional RF algorithm, the improved version has improved F1 values on all categories, especially on the categories where the performance is previously poor. This improvement is due to the optimization of the algorithm in feature processing and classification decisions. Therefore, it can distinguish between different types of cyber attacks more effectively. Figure 9 further analyzes the coverage, recall, and F1 values of the two models in different data sets scores.

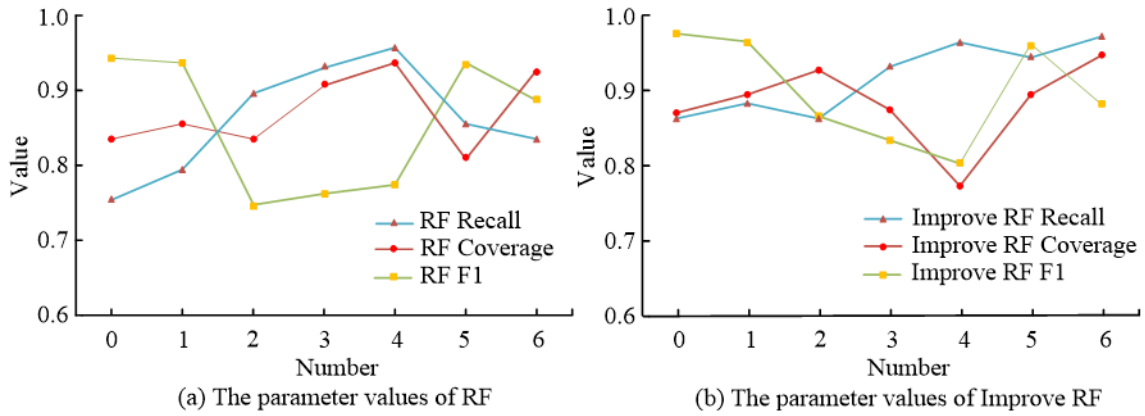

Figure 9. Comparison of values between two algorithms

Figure 9(a) represents the recall, coverage and FI values of RF. Figure 9(b) represents the recall, coverage and FI values of improved RF. From Figure 9(a), when the RF algorithm is used to detect number 0 number 7, the highest recall, coverage, and F1 value of RF are 0.96, 0.93, and 0.95, respectively. From Figure 9(b), when the improved RF algorithm is used to detect number 0 number 7, the highest recall, coverage, and F1 value of the improved RF algorithm are 0.98, 0.95, and 0.98, respectively. Figure 9 shows that the improved RF algorithm is significantly higher than the RF algorithm in classification accuracy. The test results in Figure 9 show that the improved RF algorithm has higher F1 values in categories numbered 0, 1, and 5. The improved RF algorithm model has good recognition performance for DoS/DDoS, Benign, and PortScan, and lower F1 values in number 4, indicating that the model has poor recognition performance for Infiltration. The improved RF algorithm performs better in all the metrics, especially in classification accuracy. This performance enhancement is due to the inclusion of effective feature

selection and more accurate classification decision-making mechanisms in algorithm optimization. Thus, the optimized RF has better performance in processing complex and unevenly distributed datasets.

To further confirm the effectiveness of the enhanced RF model, the string algorithm (Boyer-Moore, BM) is introduced to control the features involved in splitting. The relationship between the classification accuracy of the string search algorithm and the forest algorithm is obtained, as shown in Figure 10.

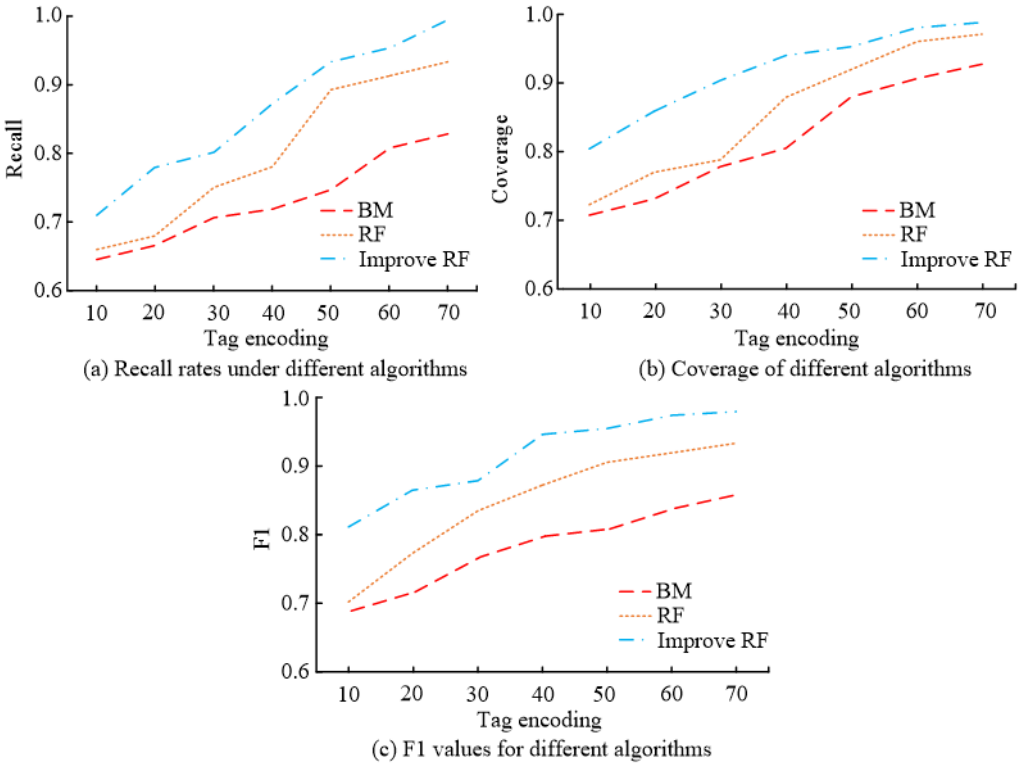

Figure 10. Number of features and accurate classification graph

Figure 10(a) represents the recall of BM string algorithm, RF and improved RF. Figure 10(b) represents the coverage of BM, RF and improved RF. Figure 10(c) represents the F1 value of BM, RF and improved RF. From Figure 10, the highest recall, coverage and F1 values possessed by the improved RF algorithm model are 0.99, 0.98, and 0.99, respectively. The algorithm outperforms the traditional RF model and the string model in all the above three metrics, which is due to the fact that the improved RF handles and utilizes the feature information more efficiently, and reduces the influence of irrelevant or noisy features, thus improving the classification accuracy.

The training time and classification time of the three algorithms are compared. Table 5 shows the calculation time for ten training classification tests on the sample set data.

Table 5. Comparison of Model Calculation Time

Type Algorithm	**BM**	**RF**	**Improve RF**
Training time	427.2s	421.7s	318.2s
Classification time	38.5s	34.7s	25.4s

From Table 5, the improved RF outperforms the RF algorithm in training and classification time. The enhanced RF training and classification time is nearly 25%, which is

better than the traditional RF algorithm. The data in Table 5 shows that the improved RF algorithm has a significant advantage in both training and classification time, due to improvements in the computational efficiency of the algorithm. For example, by reducing redundant calculations and optimizing the data processing flow, the algorithm can be trained and classified faster, especially when dealing with large-scale datasets.

5. **Conclusion.** With the development of the Internet, the complexity of the network environment is increasing. Network intrusion attacks are constantly emerging. Network intrusion detection has become a popular research problem in the field. This research improves the RF algorithm. The improved RF algorithm is combined with the intrusion detection system to identify and classify the intrusion attacks, protecting the user network security. The results show that the F1 values of all types of labels on the RF training dataset are 0.964, 0.963, 0.774, 0.778, 0.762, 0.953, and 0.871. The macroF1 value of RF algorithm is 0.867. The F1 values of all types of labels on the improved RF training dataset are 0.984, 0.982, 0.875, 0.842, 0.796, 0.982, and 0.889. The macroF1 value of improved RF is 0.907. The training time and classification time of the two algorithms are compared. The RF algorithm has a training time of 421.7s and a classification time of 34.7s, while the improved RF are 318.2s and 25.4s, which has better performance. The study uses stratified feature random sampling. Although it can improve the overall classification performance of the model, the feature selection range is reduced. Therefore, the correlation between decision trees increases and the generalization ability of the model decreases. The change in model performance under different sampling methods should be further investigated subsequently. Despite the results achieved in this study, there are still several issues and challenges that need to be addressed. Firstly, although this research has achieved improvements in the accuracy and efficiency of network intrusion detection, the detection effectiveness of certain complex attack types, such as Advanced Persistent Threats (APTs) and zero-day attacks, still needs to be improved. Secondly, current research focuses on the efficiency and accuracy. Future work can explore how to further optimize the algorithm to improve the adaptability to new and complex attacks. Finally, future research should also focus on exploring intrusion detection strategies in large-scale distributed network environments. With the development of IoT and cloud computing technologies, network environments have become more complex and dynamic. This requires IDS to adapt to constantly changing network conditions and respond promptly to emerging security threats. Therefore, more intelligent and adaptive network security mechanisms will become an important trend in the field of network security.

Acknowledgment. The research is supported by: The sixth "333 high-level talent training project" in Jiangsu Province (No.: (2022)3-18-169).

REFERENCES

[1] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Computer Networks*, vol. 174, no. 19, pp. 1072-1089, 2020.
[2] A. Nazir, and R. A. Khan, "A novel combinatorial optimization based feature selection method for network intrusion detection," *Computers & Security*, vol. 102, no. 4, pp. 1021-1027, 2020.
[3] Y. Guo, Z. Mustafaoglu, and D. Koundal, "Spam detection using bidirectional transformers and machine learning classifier algorithms," *Journal of Computational and Cognitive Engineering*, vol. 2, no. 1, pp. 5-9, 2022.
[4] M. M. Althobaiti, K. Kumar, D. Gupta, S. Kumar, and R. F. Mansour, "An intelligent cognitive computing based intrusion detection for industrial cyber-physical systems," *Measurement*, vol. 186, no. 2, pp. 115-124, 2021.

[5] K. Devi, and B. Muthusenthil, "Intrusion detection framework for securing privacy attack in cloud computing environment using DCCGAN-RFOA," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 9, pp. 4561-4583, 2022.
[6] F. Zhang, H. Kodituwakku, W. Hines, and J. Coble, "Multilayer data-driven cyber-attack detection system for industrial control systems based on network, system, and process data," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4362-4369, 2019.
[7] J. Ning, J. Wang, J. Liu, and N. Kato, "Attacker identification and intrusion detection for in-vehicle networks," *IEEE Communications Letters*, vol. 23, no. 11, pp. 1927-1930, 2019.
[8] G. O. Anyanwu, C. I. Nwakanma, J. M. Lee, and D. S. Kim, "Novel hyper-tuned ensemble random forest algorithm for the detection of false basic safety messages in internet of vehicles," *ICT Express*, vol. 9, no. 1, pp. 122-129, 2023.
[9] S. Qin, M. Zhang, X. Ma, and M. Li, "A new integrated analytics approach for wind turbine fault detection using wavelet, RLS filter and random forest," *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*, vol. 45, no. 3, pp. 8285-8300, 2023.
[10] R. Haffar, D. Sanchez, and J. Domingo-Ferrer, "Explaining predictions and attacks in federated learning via random forests," *Applied Intelligence*, vol. 53, no. 1, pp. 169-185, 2023.
[11] M. Ge, N. F. Syed, X. Fu, Z. Baig, and A. Robles-Kelly, "Towards a deep learning-driven intrusion detection approach for Internet of Things," *Computer Networks*, vol. 186, no. 26, pp. 1-11, 2021.
[12] S. N. Mighan, and M. Kahani, "A novel scalable intrusion detection system based on deep learning," *International Journal of Information Security*, vol. 20, no. 3, pp. 387-403, 2021.
[13] Kamaldeep, M. Malik, M. Dutta, and J. Granjal, "IoT-Sentry: A cross-layer-based intrusion detection system in standardized Internet of Things," *IEEE Sensors Journal*, vol. 21, no. 24, pp. 28066-28076, 2021.
[14] B. Gao, B. Bu, W. Zhang, and L. Xiang, "An intrusion detection method based on machine learning and state observer for train-ground communication systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 6608-6620, 2022.
[15] T. Saba, T. Sadad, A. Rehman, Z. Mehmood, and Q. Javaid, "Intrusion detection system through advance machine learning for the internet of things networks," *IT Professional*, vol. 23, no. 2, pp. 58-64, 2021.
[16] Y. H. Lin, B. H. Zheng, and L. Wang, "Cascaded fiber-optic interferometers for multi-perimeter-zone intrusion detection with a single fiber used for each defended zone," *IEEE Sensors Journal*, vol. 21, no. 9, pp. 10685-10694, 2021.
[17] G. Xie, L. T. Yang, Y. Yang, H. Luo, R. Li, and M. Alazab, "Threat analysis for automotive CAN networks: A GAN model-based intrusion detection technique," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4467-4477, 2021.
[18] Z. S. Khalafi, M. Dehghani, A. Khalili, A. Sami, N. Vafamand, and T. Dragicevic, "Intrusion detection, measurement correction, and attack localization of PMU networks," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 5, pp. 4697-4706, 2022.
[19] J. Wang, Z. Tian, M. Zhou, J. Wang, X. Yang, and X. Liu, "Leveraging hypothesis testing for CSI based passive human intrusion direction detection," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 8, pp. 7749-7763, 2021.
[20] H. Liu, S. Zhang, P. Zhang, X. Zhou, X. Shao, and G. Pu, "Blockchain and federated learning for collaborative intrusion detection in vehicular edge computing," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 6, pp. 6073-6084, 2021.