IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 13, 2021, accepted January 22, 2021, date of publication January 28, 2021, date of current version February 9, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3055209

Anomaly Detection and Attack Classification for Train Real-Time Ethernet

RUIFENG DUO[1], XIAOBO NIE[1], NING YANG[2], CHUAN YUE[1], (Student Member, IEEE), AND YONGXIANG WANG[2]

[1]School of Electrical Engineering, Beijing Jiaotong University, Beijing 100044, China
[2]Locomotive and Car Research Institute, China Academy of Railway Sciences Corporation Ltd., Beijing 100044, China

Corresponding author: Xiaobo Nie (xbnie@bjtu.edu.cn)

ABSTRACT Real-time Ethernet has been applied to train control and management system (TCMS) of 250km/h Fuxing Electric Multiple Units (EMUs) and some urban rail vehicles. The openness of the Ethernet communication protocol poses a risk of intrusion attacks on the train communication network. It is, therefore, necessary that a safety protection technology is introduced to the train communication network based on real-time Ethernet. In this paper, a train communication network intrusion detection system based on anomaly detection and attack classification is proposed. Firstly, the paper built an anomaly detection model based on support vector machines (SVM). The particle swarm optimization-support vector machines (PSO-SVM), and genetic algorithm-support vector machines (GA-SVM) optimization algorithms are used to optimize the kernel function parameters of SVM. Secondly, the paper built two attack classification models based on random forest. They are iterative dichotomiser3 (ID3) and classification and regression tree (CART). And then, the built intrusion detection and attack classification model is tested by using the public data set knowledge discovery and data mining-99(KDD-99) and the data set of the simulation train real-time Ethernet test bench. PSO-SVM improves the intrusion detection accuracy from 90.3% to 95.75%, GA-SVM improves the detection accuracy from 90.3% to 95.85%. The training time of the PSO-SVM algorithm was higher than that of the GA-SVM algorithm, and much higher than that of the SVM, without optimization. Both ID3 and CART models are verified valid in the attack classification, while the ID3 algorithm obtained 100% accuracy on the training set, and only 32.89% accuracy on the test set, ID3 has a poor classification accuracy of the data outside of the training set. Also, the classification time is very long for ID3 compared with CART. So the comprehensive experimental results show that the intrusion detection system of train real-time Ethernet can use the GA-SVM model for detection of abnormal data. After passing the normal data, the CART model can be used to distinguish between the types of attacks to better complete subsequent responses and operations. Compared with the anomaly detection model based on SVM, the proposed model improves intrusion detection accuracy. And the proposed attack classification algorithm based on CART can improve the computing speed while ensuring the precision of classification.

INDEX TERMS Train communication network, real-time Ethernet, intrusion detection system, attack classification.

I. INTRODUCTION

With the advent of intelligent train control and management system, more and more sensors and equipment are connected to TCMS, the data transmission in TCMS is increasing rapidly, so real-time Ethernet with a high transmission rate is introduced into TCMS. The openness of the train real-time Ethernet protocol makes TCMS vulnerable to adversary attacks. These attack methods such as port scanning, DoS attacks, and IP address spoofing may also be used in TCMS. Therefore, the introduction of Ethernet brings a threat to TCMS.

Cyber-physical systems (CPSs) security has become a critical research topic to avoid key security threats faced by these applications [1]. TCMS is one kind of CPSs which are an integral system featuring strong interactions between its cyber and physical components. TCMS security requires a different strategy from traditional information technology

The associate editor coordinating the review of this manuscript and approving it for publication was Liehuang Zhu.

(IT) security. TCMS security plays a crucial part in ensuring the normal operation of the train. Once the train data and system suffer attacks, it may cause disruption to the operation, delay to the train, and even safety accidents, thus resulting in property damage and casualties. In 2003, the train signal system in Florida suffered attacks from the ''SOBIG'' virus, with some trains forced into delay [2]. In 2008, a subway track signal in Poland was subject to attack. The attacker exploited remote control to change the track switch, thus causing derailment to four cars [3]. In 2012, the information release system and operation scheduling system of a subway in Shanghai were targeted for attack. A large number of events indicate that the security of TCMS is worthy of more attention.

To guard against potential attacks, it is essential to introduce security protection technology to defend TCMS. Representing an efficient protection technology, intrusion detection technology (IDS) is capable of identifying and judging the abnormality in the network before making a response in a targeted manner [4]. As a defense mechanism for the TCMS, intrusion detection technology performs the function of defense against attack and protection for the TCMS, to ensure the normal functioning of the train operation.

In this paper, the design of the intrusion detection model in security protection technology and the parameter optimization problem of the intrusion detection model is studied in depth. In summary, our work in this paper makes the following contributions:

1) Taking into account the characteristics of the train's real-time Ethernet, the paper divides the train intrusion detection model into two modules: anomaly detection and attack classification. Considering a large amount of real-time data in TCMS, this paper models anomaly detection as a two-class classification problem with unbalanced samples. Three anomaly detection models based on support vector machines are designed, and their performance is analyzed and compared. Considering that identifying different attacks is helpful for subsequent response and processing, these paper models the attack classification as a multiclass classification problem. Unlike support vector machines, random forests employ decision trees as individual learners which can better solve multiclass classification problems. This paper designs attack classification models based on the CART algorithm and ID3 algorithm under the random forest category.
2) The paper not only used KDD-99 to test the intrusion detection model designed in this paper but also built a simple train communication network simulation platform and collected data. Use the data set of the simulation platform to perform experimental analysis and verification of the model.

The rest of this paper is organized as follows: In section II, the paper reviewed the related work of IDS in other scenarios, briefly introduced the key component of IDS, and abstracted the problem model. In Section III, the anomaly detection model is designed, and particle swarm optimization and genetic algorithm are used to optimize the model parameters. Starting from the decision tree and the ensemble learning framework, an attack classification model based on random forest is designed in Section IV. Experiments and results are presented and discussed in Section V. Finally, the conclusion and prospect are given in Section VI.

II. RELATED WORK AND PROBLEM MODELING

This section begins with a brief review of intrusion detection systems and machine learning. Then the IDS for train real-time network is introduced. Finally, the problem modeling is carried out and the appropriate algorithm was chosen.

A. ATHE APPLICATION OF IDS IN INDUSTRIAL SCENE AND ITS COMBINATION WITH MACHINE LEARNING

Representing an efficient protection technology, intrusion detection technology is capable of identifying and judging the abnormality in the network before making a response in a targeted manner [5]. IDS is referred to as a technology that monitors network data in real-time and issues an alarm when suspicious data is detected. Back in 1980, Anderson first proposed the concept of Intrusion Attempt [6]. In 1986, Dorothy pioneered in establishing an intrusion detection system model [7]. At present, intrusion detection technology has been commonly applied in different fields. More importantly, it continues the integration of new information technology to maintain development. Based on the characteristics that DDoS attacks will cause the wavelet variance to change, literature [8] applies time-frequency analysis methods to detect network traffic attacks. In literature [9], an analysis is carried out of the communication mode, network topology, and functional information in industrial networks, intrusion behavior features are extracted, detection rules are established, and then Markov models are applied to classify them. Literature [10] collects the historical data on industrial control networks and distinguishes intrusion behaviors through signal difference analysis, based on which a new type of detector is designed to identify the hidden attacks on the power grid. In literature [11]–[14], both studies and analyses are conducted of the relevant data of industrial systems such as generators, water treatment systems, numerical control processing systems, hot blast stoves, and so on. Besides, a normal data model is constructed, with partial least squares, KNN, and other algorithms applied for abnormal detection. The above-mentioned intrusion detection technologies have achieved satisfactory results in their respective fields. As the type of attack diversifies and the intensity of intrusion increases, however, the existing intrusion detection technology has encountered such problems as increased false alarm rate and reduced detection speed.

Artificial intelligence technology has also promoted the development of intrusion detection technology to some extent, expanded its application field and detection depth, and improved the detection efficiency of the system. At present,

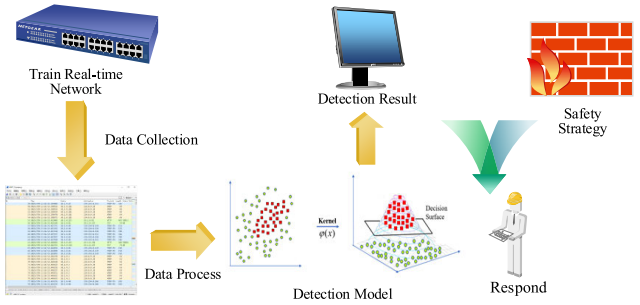


FIGURE 1. Intrusion detection process.

there are plenty of intrusion detection studies conducted to introduce machine learning methods. Literature [15] proposes a method called Isolation Forest (iForest), which detects anomalies purely based on the concept of isolation without employing any distance or density measure. Literature [16] puts forward an intrusion detection model integrating chi-square feature selection and multi-class support vector machine. Literature [17] proposes the model which is an integrated intrusion detection system combining Chi-square as the feature selection technique and a hybrid model of base and ensemble classifiers. In literature [18], an integrated model combined with multiple classifiers is adopted to detect DDoS attacks, which leads to desirable results on the NSL-KDD data set. Based on the introduction of whale algorithms for optimization, literature [19] combines network intrusion detection with Support Vector Machine, thus improving the accuracy of intrusion detection. Literature [20] constructs a fingerprint recognition framework that relies on lightweight signatures to filter out abnormal data by routers. Literature [21] applies a three-layer neural network to train the KDD-99 data set, with the anomaly detection result reaching a 95% accuracy. Despite these detection methods combined with machine learning has achieved relatively satisfactory experimental results, their algorithms remain subject to certain limitations. For example, the premature convergence of the genetic algorithm, the problem with parameter selection in the SVM algorithm, and the problem with neural network training data set. Moreover, for other algorithms, the convergence speed is excessively slow, which makes it easy to fall into local optimization. These problems need to be addressed by improving the algorithm. Some key indicators to measure the intrusion detection model are presented in the literature [22].

B. INTRUSION DETECTION SYSTEM FOR TRAIN REAL-TIME ETHERNET

As exhibited in Figure 1, the intrusion detection system (IDS) for train real-time Ethernet is mainly divided into three parts. The first part is information collection, which mainly collects system log information or network traffic data from the switch of train critical nodes. The second part is data analysis, which processes the collected data, establishes a detection model, and identifies intrusion behaviors. The third part is the system response, suggesting that the system can take corresponding actions. In the above three parts, the data analysis module is the core of the intrusion detection system, and the intrusion detection technology is the core of the data analysis module. In this paper, we mainly study the data analysis module.

C. PROBLEM MODELING AND CORE ALGORITHM SELECTION

Although there has been a lot of research on intrusion detection systems, there is still a gap in the research on intrusion detection systems for train real-time Ethernet. The literature research technical analysis mentioned above makes it clear that the main object of this study is real-time Ethernet for trains, and the main research core is intrusion detection algorithms. This study aims to recognize normal data and abnormal data. It has been revealed that the intrusion detection system is essentially a pattern recognition problem, that is, a classification problem. It requires the ability to accurately analyze what category the input data belongs to. On this basis, the problem model is established as follows:

$$\max K = \omega_1 \text{Accuracy} + \omega_2 \text{Spacificity} \tag{1}$$

$$st.\ T(n) < \bar{t} \tag{2}$$

$$Sensitivity \geq A_3 \tag{3}$$

where, *Accuracy*— the proportion of correctly classified samples in the total sample (%);

Spacificity— the identification of abnormal data (%);

Sensitivity— the proportion of all positive examples sorted (%).

Constraint (2) denotes the algorithm running time constraint; Constraint (3) suggests the normal data passing constraint.

Considering various algorithm characteristics and train real-time Ethernet intrusion detection requirements, the intrusion detection problem is divided into two parts in this article. In the first part of the model, the binary classification of normal data and abnormal data can be realized. It is based on a support vector machine for anomaly detection and can intercept abnormal data because the support vector machine has high accuracy and strong generalization ability for binary classification problems and can deal with the sample imbalance of normal data and abnormal data. In the second part of the model, abnormal data can be multi-classified to distinguish different attack types. It classifies abnormal data based on the random forest to identify different attack types since support vector machines perform poorly on multi-classification problems, the random forest can handle multi-classification problems well, and the running time is short.

III. ANOMALY DETECTION MODEL BASED ON SUPPORT VECTOR MACHINE

Support vector machine can better process high-dimensional and nonlinear data sample classification problems and has a high classification accuracy and strong generalization ability [23], so it can solve the problem of sample imbalance.

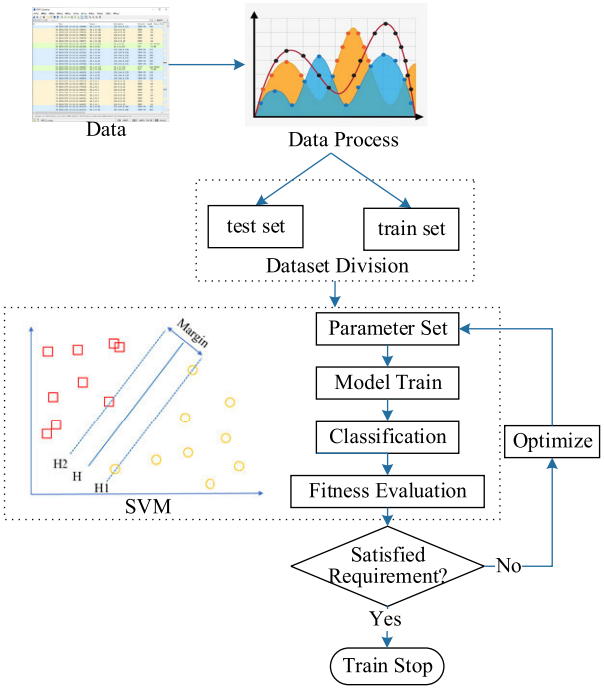


FIGURE 2. Anomaly detection process based on SVM.

Therefore, in this section, the intrusion detection model is designed based on SVM. Then particle swarm optimization and genetic algorithms are used to search and optimize parameters.

A. ADESIGN OF ANOMALY DETECTION MODEL BASE ON SUPPORT VECTOR MACHINE

Given that the data $X = \{X_1, \ldots, X_N | X_i \in \mathrm{R}^d\}$ is linearly inseparable, assuming that there is a nonlinear mapping $\varphi : X \to H, X \in R^d, H \in R^k$, it will be mapped from the Euclidean space to the Hilbert space H, making its data set linearly separable in space H.

$$\max \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \varphi(x_i^T) \varphi(x_j) \tag{4}$$

$$s.t. \sum_{i=1}^{N} \alpha_i y_i = 0 \tag{5}$$

$$0 \le \alpha_i \le C, \quad i = 1, .., N \tag{6}$$

According to the above problem, the kernel function can be defined as:

$$K(x, z) = \varphi(x) \bullet \varphi(z) \tag{7}$$

The kernel function defines the similarity of the two data after the mapping transformation, and then the classification is conducted by the SVM.

Radial Basis Function (RBF) kernel, also known as Gaussian kernel or Squared Exponential (SE) kernel, is frequently applied in various types of learning algorithms, such as Gaussian Process Regression (GPR) [24]. It is usually defined as the monotone function of the Euclidean distance between a given point in space to a center, that is:

$$K(x, z) = \exp(-\gamma \|x - z\|^2) \quad \gamma > 0 \tag{8}$$

In this chapter, the model design will use RBF kernel functions to map the data.

The anomaly detection model based on SVM is shown in Figure 2. The specific establishment steps are as follows:

Step 1: Collect real-time Ethernet data of the original train and perform feature statistical processing to obtain various data feature values. Then the normalization and encoding processing is performed.

Step 2: Use a simple random non-replacement sampling method to construct the training set and test set from the original data set.

Step 3: Set the kernel function and the threshold of each parameter in the training process, and take the parameter vector g as the optimization parameter.

Step 4: Construct and solve the quadratic programming problem and obtain the solution:

$$\alpha^* = \left(\alpha_1^*, \alpha_2^*, \ldots, \alpha_n^*\right)^T \tag{9}$$

Step 5: Solve ρ^* according to support vector x_i^* and α_i^*

$$\rho^* = \sum_{j=1}^{n} \alpha_j^* K(x_j, x_i) \tag{10}$$

Step 6: Construct a classification decision function:

$$f(x) = sign\left(\sum_{i=1}^{n} \alpha_i^* K(x_1, x) - \rho^*\right) \tag{11}$$

Step 7: Predict the test set by using the obtained model, input the classification results and other parameters into the fitness function, and return the obtained fitness value to the parameter optimization process.

Considering that accuracy and specificity are the main indicators for evaluating the classification results, the results are evaluated from these two aspects. The fitness functions are:

$$K = \omega_1 \cdot A_{TEST} + \omega_2 \cdot B_{TEST} \tag{12}$$

where, A_{TEST} – test set accuracy;

B_{TEST} – test set specificity;

The selection of kernel function parameters is very crucial to solve the problem of nonlinear data and improve classification accuracy. Considering that the selection of parameters based on personal experience is random, it is difficult to obtain a more ideal result. Therefore, the optimization process will optimize the search for the kernel function parameters.

B. BDESIGN OF OPTIMIZATION PROCESS BASED ON PARTIAL SWARM OPTIMIZATION

The first model design proposed in this section is the intrusion detection model design which is based on SVM and conducts parameter search by adding the PSO algorithm.

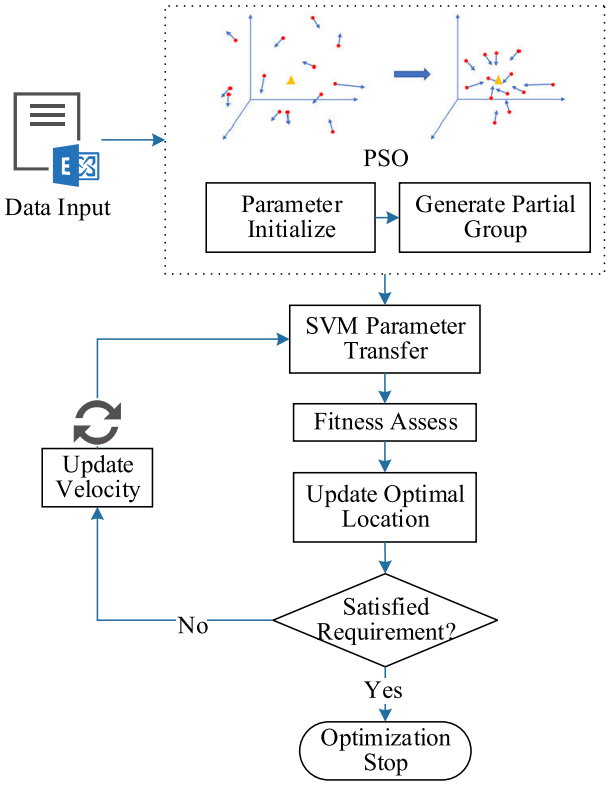


FIGURE 3. Optimization process based on PSO.

Figure 3 shows the model design, and the specific steps are as follows:

Step 1: Initialize parameters. Set the maximum iterations $\delta_{\max}$, accuracy ε, and parameter thresholds.

Step 2: Generate a particle swarm. Randomly generate a particle swarm containing N particles, and each of them contains position X and speed V. Among them, the position is the searched kernel function parameter vector g.

Step 3: Individual evaluation. The position of each particle is introduced into the SVM model as a parameter to obtain the test result of the model, then calculate the fitness function of each particle, and obtain the fitness vector $Y = f(X)$.

Step 4: Update individual optimal position and global optimal position. The specific steps are as follows:

If the adaptability of a particle's current position is greater than the individual optimal fitness, that is $f(p_{id}(t)) < f(x_i(t+1))$, then the current particle's optimal position will be updated, that is $p_{id}(t+1) = x_i(t+1)$. Otherwise, the last individual optimal position will still be maintained, $p_{id}(t+1) = p_{id}(t)$.

After the individual optimal position completes the update, it is sorted in order from large to small according to its fitness, and the optimal value is selected.

If the optimal value of the individual optimal fitness is greater than the global fitness, that is $f(p^*_{id}(t+1)) > f(p_{gd}(t))$, then the global fitness will be updated to that value, that is $p_{gd}(t+1) = p^*_{id}(t+1)$. Otherwise, the last individual optimal position will still be maintained, $p_{gd}(t+1) = p_{gd}(t)$.

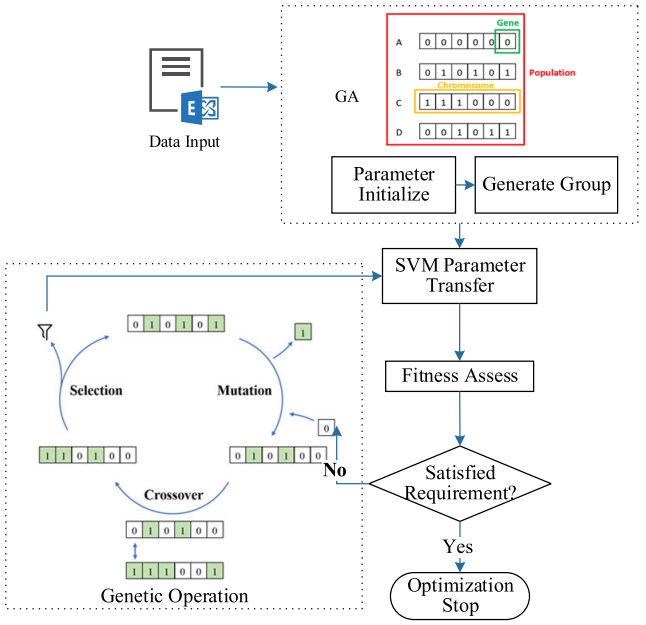


FIGURE 4. Optimization process based on GA.

Step 5: Terminate determination. Determine whether the termination condition is met, and the judgment conditions are as follows:

(1) Whether the maximum number of iterations is reached, and if so, stop the iteration and output the result.

(2) Whether the difference between the global optimal fitness for three consecutive times is within the accuracy, if it is, then stop the iteration and output the result.

If the termination judgment requirement is not met, continue to perform STEP 6.

Step 6: Update the particle speed and position according to the flight formula. If the parameter exceeds the threshold, the parameter will be set as the threshold value. After updating, go back to STEP 3 to conduct model training and fitness evaluation.

C. DESIGN OF OPTIMIZATION PROCESS BASED ON GENETIC ALGORITHM

The second model design proposed in this section is the intrusion detection model design which is based on SVM and conducts parameter searching by adding GA. Figure 4 shows the model design, and the specific steps are as follows:

Step 1: Initialize parameters. Set the evolution algebra counter, set the maximum evolution algebra $\delta_{\max}$, crossing-over rate λ, variability π, and parameter thresholds.

Step 2: Generate a population. Randomly generate a population containing N individuals and take the kernel function parameter vector g as the performance characteristics, and then the performance characteristics are chromosomally encoded according to the binary coding method.

Step 3: Individual assessment. Decode the chromosomes of each particle to get the performance characteristics. Introduce the performance characteristics, that is, parameters, into the SVM model to obtain the model detection results. The fitness

function of each individual is calculated according to the detection results, and then the fitness vector $Y = f(X)$ is obtained.

Step 4: Terminate determination. If the maximum evolutionary algebra is reached, the iteration will be terminated. If the termination judgment requirements are not met, go to STEP5 to conduct genetic manipulation.

Step 5: Genetic manipulation. Genetic manipulation includes three processes: selection, crossover, and mutation. The specific steps are as follows:

Selection: Use the roulette wheel selection method to select individuals, that is, take individual fitness as the selection probability for random selection and retain some candidate solutions.

Crossover: Use a two-point crossover method to randomly exchange genes between the parents of each parent according to the crossover rate, that is, randomly set two crossover points on the chromosome and exchange the genes at this site.

Mutation: Use the binary mutation method to randomly select some individuals for 0-1 conversion at any gene position according to the mutation rate.

After completing the genetic manipulation, turn to STEP 3 to evaluate individuals and make termination determination.

IV. ATTACK CLASSIFICATION MODEL BASED ON RANDOM FOREST

Ethernet attacks come in many varieties. A train intrusion detection model that simply detects whether the data is abnormal is not enough to trigger a response to an attack. Therefore, it is necessary to identify different types of intrusion data. This multi-classification problem can be handled by continuous two-class classification, although the effect is far less than the multi-class classification method. Random forest is an integrated learning method suitable for handling multi-classification problems, and this method can handle high-latitude data with high accuracy. At the same time, the parallel training processing method also guarantees the real-time requirements of intrusion detection.

For random forest models, the key lies in the design of decision trees and integrated learning frameworks.

A. DESIGN OF ATTACK CLASSIFICATION MODEL BASED ON DECISION TREE

The decision tree is a classic supervised machine learning approach. Commonly used algorithms are ID3 and CART [25]. In Table 1, we compare ID3 and CART.

The design of the attack classification decision tree is shown in Figure 5.

Step 1: Input data set D and initialize threshold parameters (maximum depth, minimum sample number of nodes, Gini coefficient threshold, pruning regularization threshold).

Step 2: Calculate each feature value pair data set in the current node D Gini coefficient.

Step 3: Select the feature with the smallest Gini coefficient A and corresponding eigenvalues a. According to the selected features and feature values, the data set is divided into two parts D_1 and D_2. Simultaneously, establish the left and right child nodes of the current node.

Step 4: Determine whether the current sample number of each child node is less than the minimum node sample number or no optional feature. Set the node that meets the condition as a leaf node and stop growth.

Step 5: Determine whether all end nodes are leaf nodes. If all nodes are leaf nodes, stop growing, and skip STEP 6.

Step 6: Calculate the Gini coefficient of the root node data set. If it is less than or equal to the threshold, stop the recursion and continue with STEP 7; otherwise, set the node data closest to the root node as the data set D and go back to STEP 2.

TABLE 1. Comparison of decision tree algorithms.

Type	ID3	CART
Optimal partition attribute	Information gain	Gini coefficient
Decision tree type	Multitree	Binary tree
Processing variable types	Only handle discrete variables	Discrete/continuous variables can be processed
Function	Only do classification	Regression / classification
Missing value	Sensitive to missing values	Can handle missing values
Scope of application	Suitable for small samples	Suitable for large samples

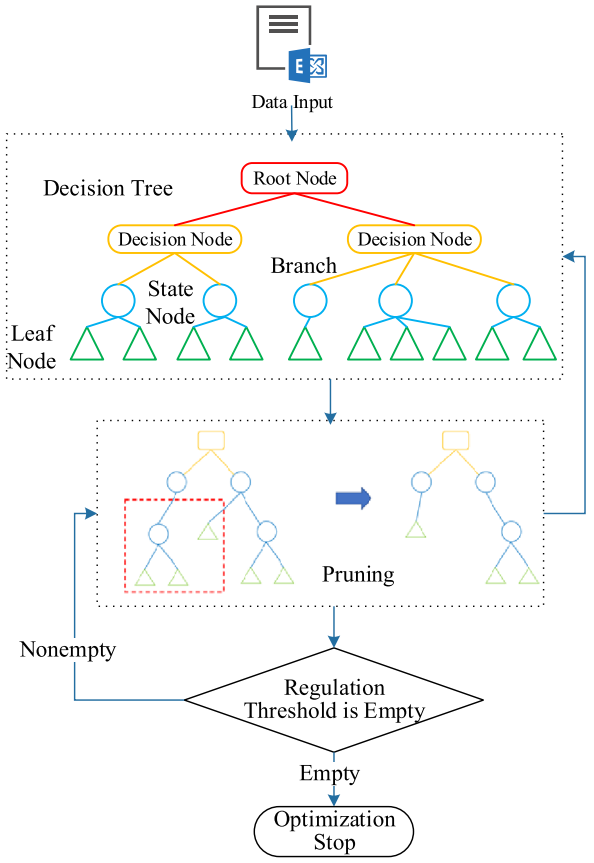


FIGURE 5. Process of decision tree.

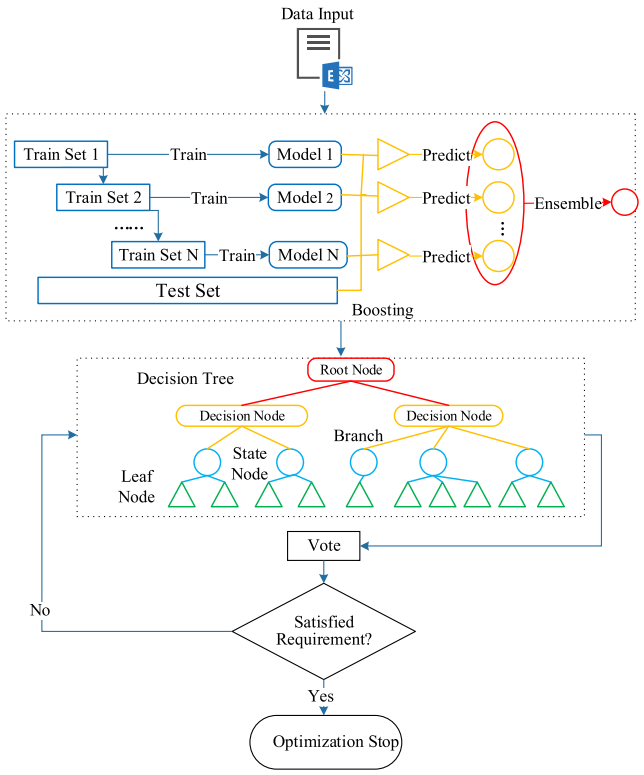


FIGURE 6. Process of random forest.

Step 7: Calculate the training error loss function $C_a(T_i)$ of each internal node from the leaf node to the bottom and regularization threshold α.

Step 8: Select the largest α_M of all nodes. Access all internal nodes from top to bottom. Determine whether pruning should be done according to the set criteria.

Step 9: Determine whether the regularization threshold set is empty, if it is empty, output the optimal decision tree; otherwise, go back to STEP 7 and prune.

B. DESIGN OF ENSEMBLE ARCHITECTURE BASED ON RANDOM FOREST

The process involves obtaining M training sets from the original data set containing N sets of data through Bootstrap, for each training set, training independent M weak learners, and output the result through the specified combination strategy. The autonomous sampling method is a sampling method with replacement, each time extracting K sets of data from the original data set, a total of Group M, thus gets M training sets.

The attack classification framework based on random forest includes the steps of sample selection, individual learning training and integration, and testing. The process is shown in Figure 6.

Step 1: Initialize frame parameters, including the number of training sets, features, and samples in a single data set.

Step 2: From the original data set X_{train} with capacity N, use Boosting to extract $M(M < N)$ samples with replacement as a training subset O_i. Repeat until the number of training sets is satisfied.

TABLE 2. Feature.

Feature				
Duration	Flag	Src_bytes	Dst_bytes	Wrong_fragment
Urgent	Hot	Num_failed_logins	Root_shell	Num_root
Num_file_creations	Num_shells	Num_access_files	Count	Srv_count

Step 3: For each training subset O_i, from the original data F, randomly select $K = \log_2 F$ (Roundup) features to generate a new training set s.

Step 4: For each training set D_i, use the model of attack classification model to generate a decision tree T_i, combining random trees to generate a random forest.

Step 5: Input the test set X_{test} into the generated random forest and use the majority vote method to obtain the test result.

Step 6: Perform test verification to determine whether the generated random forest meets the requirements.

V. EXPERIMENT ON ANOMALY DETECTION MODEL

This chapter will use the processed KDD-99[26] data set to conduct an experimental analysis on the proposed models, to verify the algorithm's ability to effectively identify normal data and attack data, and analyze the key factors and performance indicators. Moreover, the data set of the experimental platform is used to verify the model.

A. ADATA SET DESCRIPTION IN THE EXPERIMENT

Many intrusion detection models use the KDD99 data set to evaluate the performance of the model [27]–[31]. KDD-99 data set is divided into two parts: the training set contains more than 5,000,000 pieces of network connection records, and the test set contains more than 2,000,000 pieces of network connection records. Each network connection is marked as normal or attack. The attack is divided into 4 major categories, a total of 39 types, of which 22 types appear in the training set, and the other 17 types of unknown attacks appear in the test set. Each network connection of KDD-99 is described by 41 characteristics, and it can be divided into four categories, including basic characteristics of TCP connection, content characteristics of TCP connection, time-based network traffic statistical characteristics, and host-based network traffic statistical characteristics.

This experiment is based on this data set and selects 15 commonly used features for training. Also, some of the attack types that did not participate in the training are selected and put into the test sample to evaluate the algorithm's generalization ability. Among them, 15 characteristics are shown in Table 2, and the selected attack types are shown in Table 3.

* represents the attacks only appearing in the test set.

The experiment selects about 2000 data as the sample set, and 500 data as the test set. Among them, 70% is normal data and 30% is attack data, which is used to verify the algorithm's ability to settle the sample imbalanced problem.

TABLE 3. Attack type.

Attack				
Guass_password	Ipsweep	Mscan	Neptune	Nmap
Saint	Buffer_overflow	Land*	Load module*	Multihop*
Named*	Perl*	Phf*	Pod*	Portsweep*
Processtable*	Worm*	Rookit*	Satan*	Smurf*

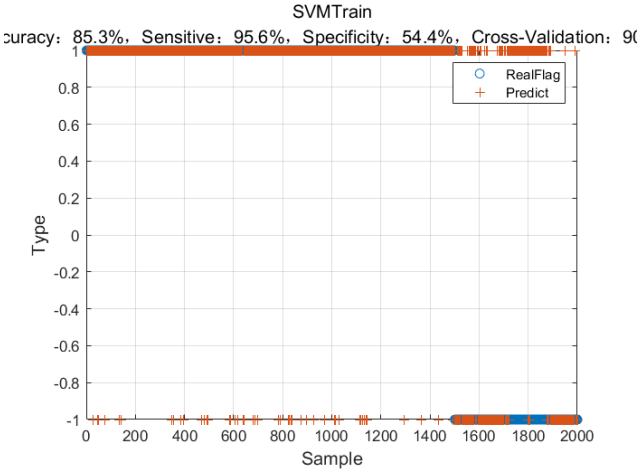


FIGURE 7. SVM on the train data set.

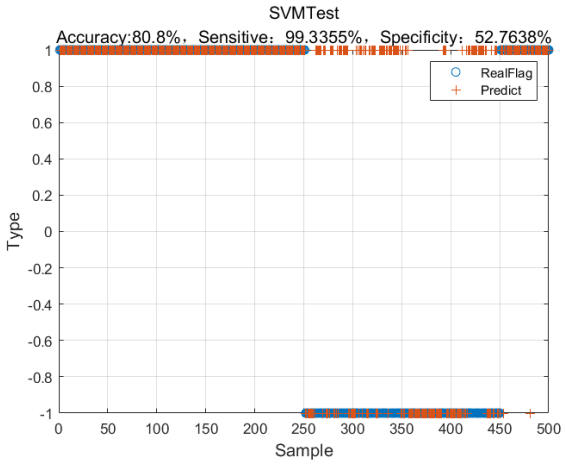


FIGURE 8. SVM on the test data set.

B. EXPERIMENT ON ANOMALY DETECTION MODEL

The computer environment used in the experiment is the following operating system: CPU: Intel (R) Core (TM) i5-6300 U, main frequency: 2.40GHz, memory: 8GB, and Windows 10 64-bit. Then the simulation software MATLAB 2018a is used to experiment on the proposed algorithm.

First of all, verify the detection results based on SVM without optimization. Set the parameters as follows: $C = 1$, $G = 1$, the cross-check uses the K-fold method, K is set as 5. The results are shown in Figures 7 and 8 shows:

According to the test results, it can be concluded that the accuracy of the training set is 92.9%, the sensitivity is 99.6%, the specificity is 72.8%, and the cross-check result is 92.7%; the accuracy of the test set is 82.4%, the sensitivity is 99.67%, and the specificity is 56.2814%, the cross-check result is 87.6%. Based on the results, it can be found that the training set detection results of the unoptimized support vector machine algorithm meet the requirements, and the accuracy of the test set still needs to be improved. Besides, it can also be directly seen from the figure that the model's identification ability for the negative class, that is, specificity, is relatively poor and needs to be improved.

Later, traverse the parameters c and g within the range of 1-100 to observe the performance index of SVM, and the results are shown in Figure 9.

The running time of the experiment is 1393s. Through experiments, it can be found that the accuracy and specificity also continuously change with the parameters, and the variation trend of the accuracy and specificity is generally the same, that is, there is an optimal parameter combination, which makes the abnormality detection performance based on SVM optimal. Considering that the manually selecting parameter has certain randomness, and the running time for traversing the parameters is too long, it is necessary to introduce PSO and GA algorithms to optimize the parameters based on SVM.

First of all, verify the results obtained by optimizing the parameters with PSO. The basic settings are shown in Table- 4.

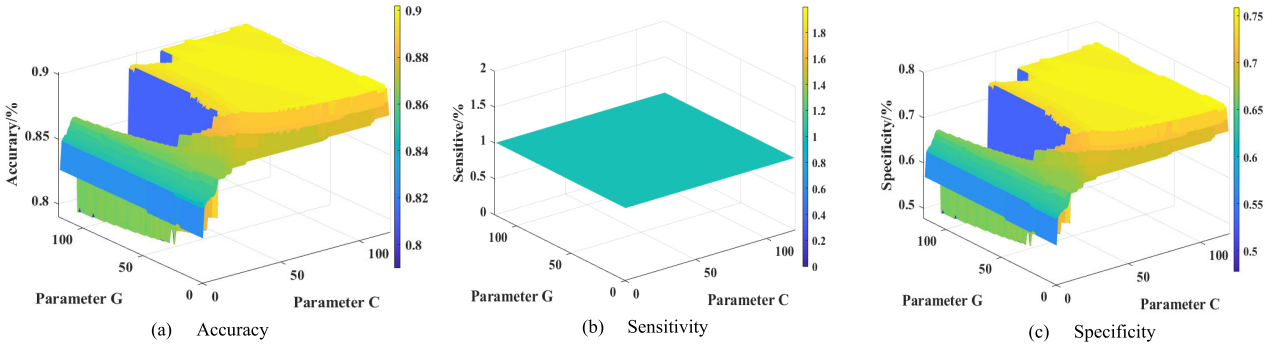
The experimental results are shown in Table 5. Based on the results, it can be found that the accuracy and specificity performance of the intrusion detection system optimized by the PSO algorithm is significantly improved. Besides, the experiment proves that PSO-SVM can meet the requirements.

Secondly, verify the results of the results obtained by optimizing the parameters with GA. The basic settings are shown in Table 6.

The experimental results are shown in Table 7. Based on the results, it can be found that the accuracy and specificity performance of the intrusion detection system optimized by the GA algorithm is significantly improved. Compared with the PSO algorithm, the GA algorithm has higher accuracy, sensitivity, and specificity in both the training set and the test set after optimization. Only cross-validation is slightly lower, and the GA algorithm has better overall performance after.

The fitness curves of the two optimization algorithms are shown in Figures 10 and 11. According to the changes in fitness, it can be found that the optimization speed of the PSO algorithm is slightly faster, but the stability of the average fitness is slightly worse. Although GA requires more algebras, its average fitness is relatively stable.

Change the number of sample inputs and observe the changes in the running time of the three algorithms. The experimental results are shown in Figure 12.



(a) Accuracy (b) Sensitivity (c) Specificity

FIGURE 9. Traverse the parameter.

TABLE 4. PSO-SVM Parameter.

Parameter	Value	Explanation
C1	0.5	Local search capability
C2	0.7	Global search capability
Maxgen	10	Maximum generation algebra
PopSize	10	Population size
Wv	0.5	Velocity elastic coefficient
Wp	0.5	Elasticity of position
C Range	0.01-1000	Parameter C range
G Range	0.01-1000	parameter G range

TABLE 5. PSO-SVM vs SVM.

Type	SVM training	SVM test	PSO-SVM training	PSO-SVM test
Accuracy	92.90%	82.40%	95.75%	90.00%
Sensitivity	99.60%	99.67%	99.67%	99.67%
Specificity	72.80%	56.28%	84.00%	75.38%
Cross Validation	92.70%	87.60%	92.80%	94.60%

TABLE 6. GA-SVM Parameter.

Parameter	Value	Explanation
Maxgen	10	Maximum generation algebra
PopSize	10	Number of individuals
Ggap	0.9	Population screening probability
C Range	0.01-1000	Parameter C range
C Range	0.01-1000	Parameter C range
G Range	0.01-1000	Parameter G range

It can be found that the running time of GA-SVM and PSO-SVM models is linearly proportional to the sample size, and the running time of the unoptimized SVM model does not change with the sample size. Besides, it can also be found that the PSO-SVM algorithm needs the longest running time, followed by GA-SVM, and the unoptimized SVM algorithm requires the shortest running time.

TABLE 7. GA-SVM vs SVM.

Type	SVM training	SVM test	GA-SVM training	GA-SVM test
Accuracy	92.90%	82.40%	95.85%	90.60%
Sensitivity	99.60%	99.67%	99.80%	99.67%
Specificity	72.80%	56.28%	84.00%	76.88%
Cross Validation	92.70%	87.60%	92.95%	93.40%

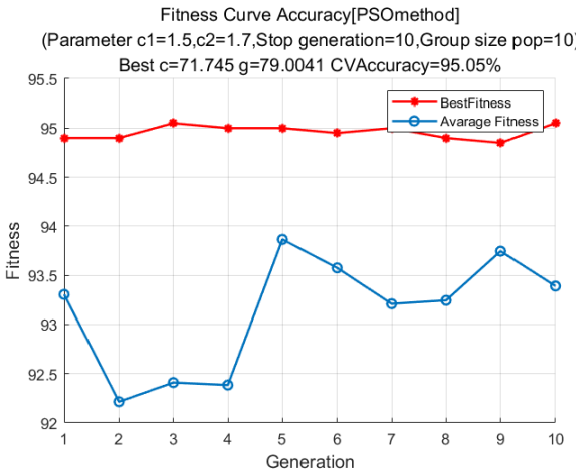


FIGURE 10. PSO-SVM fitness curve.

C. EXPERIMENT ON ATTACK CLASSIFICATION MODEL

The experiments verified the model's ability to recognize different types of attacks. The experimental environment is the same as the anomaly detection model experiment based on SVM. Experiment from KDD-99 data sets randomly select 6 types of attacks. Each attack selects 500 data to form a data set, making up a total of 3000 data. Among them, 60% is used for training and 40% is for testing. Considering that our goal is attack classification, the fitness function is adjusted to:

$$K = \sum_{i=1}^{m} \omega_i \cdot A_i \tag{13}$$

where, A_i— the recognition accuracy rate of various attacks;

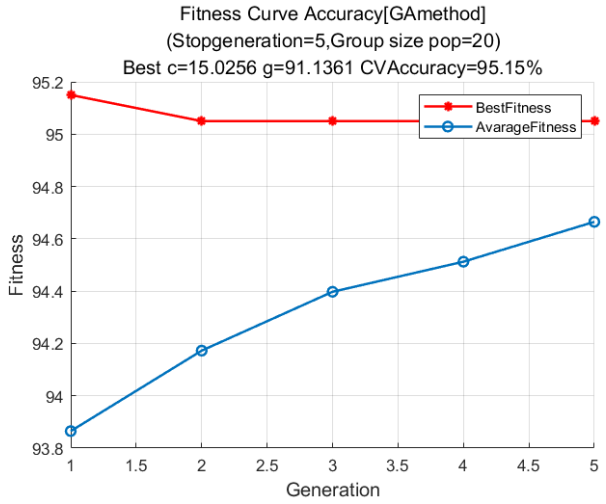


FIGURE 11. GA-SVM Fitness curve.

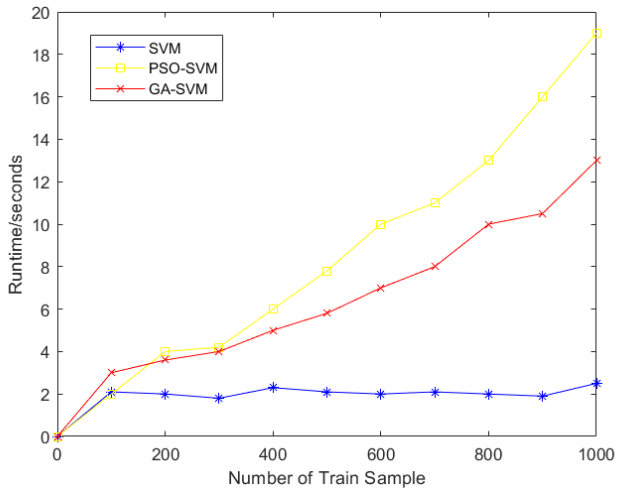


FIGURE 12. Runtime variation with the number of input data.

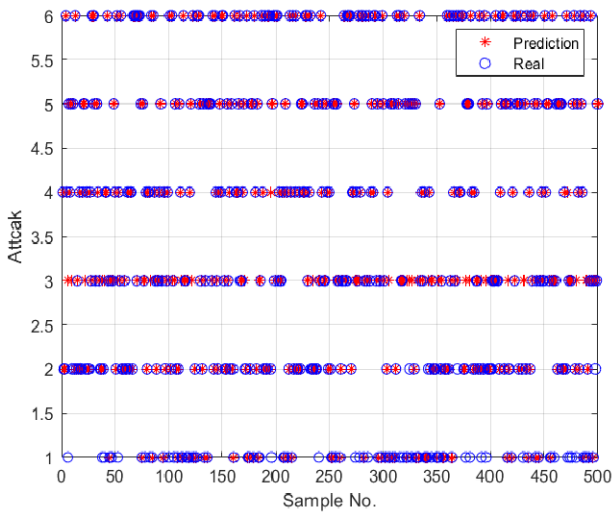


FIGURE 13. ID3 model on the test set.

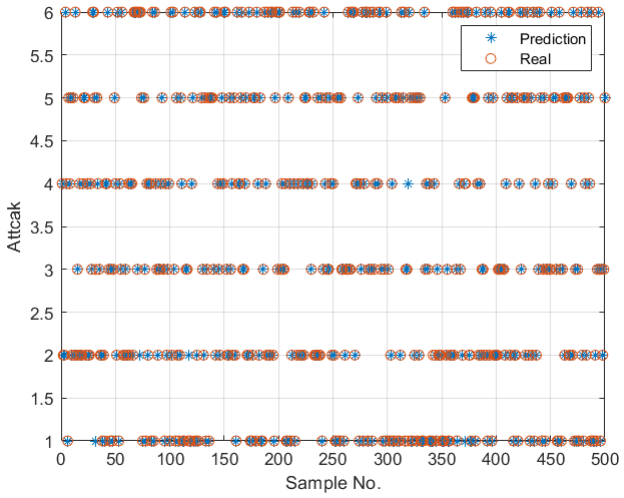


FIGURE 14. CART model on the test set.

ω_i— the attack weight factor. In this experiment, $\omega_i = 1/m$.

First, verify the ID3 model. ID3 models use information entropy as a tree standard. The experimental results show in Figures 13.

The accuracy of the ID3 model in the training and test sets are 100% and 92.58%, respectively. We can observe that the training set features are more evident, the reason is the decision tree in the ID3 algorithm is fully grown and does not perform pruning. The algorithm showed overfitting with some samples. The CART model uses information entropy as a tree standard. The results show in Figures 14.

The accuracy of the CART model in the training and test sets was 99.88% and 99.16%, respectively. We found that CART was faster while its accuracy on the test set compared to the ID3 model was improved.

Adjust the maximum number of splits for decision-making and observe the generation process of the tree. The results show in Table 8.

The number of decision points increases along with the depth of the decision tree. The accuracy of the training and test sets increases along with the complexity of decision-making. When the maximum number of divisions approaches the final number of classifications, the accuracy rate changes greatly. When the depth of the decision tree or the number of decision points reaches a higher value, the accuracy increase is small or null. Therefore, when setting the initial parameters of the decision tree, it is not necessary to account for the final depth of the decision tree. To ensure accuracy, the depth of the decision tree can be appropriately reduced, thereby reducing the complexity of the tree and the arithmetic operation time. Finally, change the number of sample inputs and observe the variations in the running times of the two algorithms. The results shown in Figure 15, reveal that these two algorithms possess a constant time complexity and that the running time of the ID3 model is slightly longer than that of the CART model.

TABLE 8. Decision tree generation.

Split Numbe	Training set	Test set	Split Numb	Training set	Test set
1	34.94%	30.75%	9	97.00%	96.00%
2	51.56%	47.08%	10	98.11%	97.00%
3	68.11%	63.75%	11	98.17%	97.08%
4	92.7%	87.6%	12	98.22%	97.25%
5	85.78%	83.67%	13	99.11%	98.67%
6	96.78%	95.75%	14	99.28%	98.92%
7	96.89%	95.83%	15	99.33%	98.92%
8	97.00%	96.00%	16	99.33%	98.92%

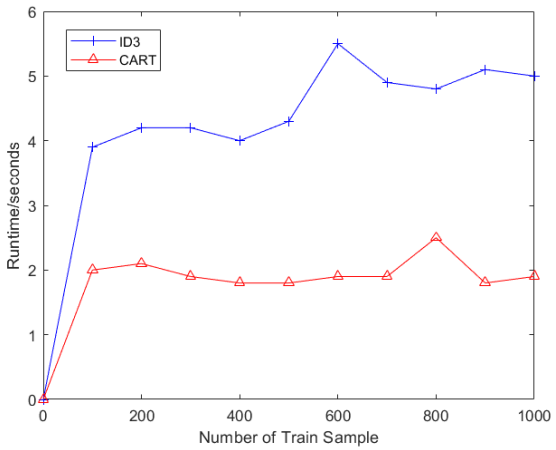


FIGURE 15. Runtime variation with the number of input data.

D. TRAIN REAL-TIME ETHERNET INTRUSION DETECTION VERIFICATION EXPERIMENT

In this section, a simple train network simulation platform is built and data is collected to perform experimental analysis and verification on the aforementioned design model. We use a switch to connect the network management system (NMS) and the end device (ED) to build a simple network. The specific network structure is shown in Figure 16.

Considering the possible characteristics of the attack and the key parts of the message, this experiment selects ten features: Time, Source Address, Source Port, Destination Address, Destination Port, Length, Cumulative Bytes, Protocol, Information, Time Since First Frame. Natural numbers are used to encode non-numeric types, and the extreme value method was used to normalize all data. First, we verified the ability of the anomaly detection model to detect abnormal data. The SVM, PSO-SVM, GA-SVM algorithm training models were used on the new training set and verified on the training set and test set respectively. The results are shown in Table 9.

The training times used by the three algorithms, SVM, PSO-SVM, and GA-SVM, were 5.006s, 88.045s, and 47.356s respectively. The training results obtained by GA-SVM and PSO-SVM were better than the SVM, without optimization on both the test set and the training set. From that time, the GA-SVM algorithm time was reduced by half compared to the PSO-SVM time. While the SVM algorithm did not require an optimization process, its required time was significantly shorter.

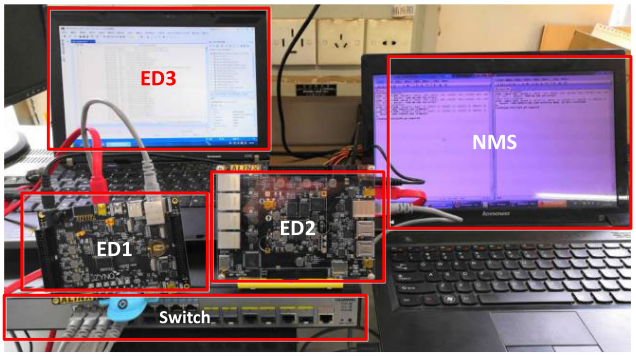


FIGURE 16. The topology of train real-time Ethernet experiment platform.

TABLE 9. Train data set on anomaly detection model.

Type	SVM test	PSO-SVM test	GA-SVM test
Accuracy	83.90%	98.80%	98.70%
Specificity	86.82%	99.04%	97.36%
Cross Validation	85.58%	98.26%	98.96%

TABLE 10. Classification model's accuracy.

Type	CART 's Accuracy	ID3 's Accuracy
Test set	96.20%	32.89%
Train set	99.26%	100.00%

Next, we verified the classification ability, of the random forest algorithm, of abnormal data. On the new training set, using the CART, the ID3 algorithm generated a random forest and verified them on the training set and the test set respectively, as shown in Table 10. The total time used by the CART algorithm was 2.161s. The ID3 algorithm spanning-tree time and classification were 2.69s and 6.753s respectively. The CART algorithm achieved 95% accuracy on the training and test sets. The ID3 algorithm obtained 100% accuracy on the training set and only 32.89% accuracy on the test set. ID3 has a poor classification accuracy of the data outside of the training set. Besides, the classification time is very long for ID3 compared with CART.

Because the amount of data is constantly increasing during the actual train operation, the number of training samples can be continuously increased. Study the relationship between the operation time of the above five models with the change in the number of samples.

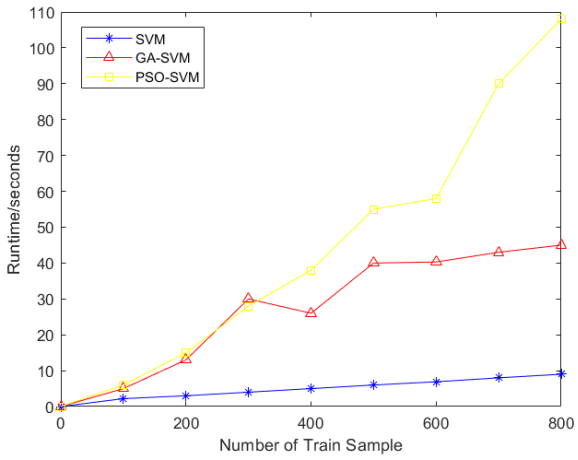


FIGURE 17. Anomaly detection model runtime.

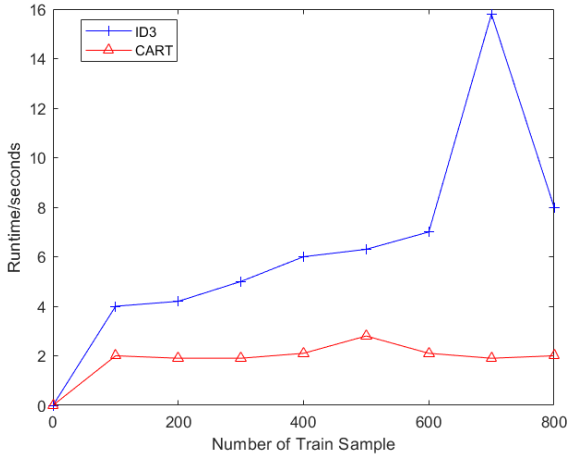


FIGURE 18. Attack classification model runtime.

The experimental results of the anomaly detection algorithm are shown in Figure 17. Observe that as the number of samples increases, PSO-SVM and GA-SVM training time increases linearly. The SVM, without optimization, maintains a constant time complexity. You can see that the training time of the PSO-SVM algorithm was higher than that of the GA-SVM algorithm, and much higher than that of the SVM, without optimization.

The experimental results of the attack classification algorithm are shown in Figure 18. It is observed that as the number of samples increased, although the training time fluctuates, it was maintained at a relatively short and constant time. Simultaneously, the training duration used by the CART algorithm was less than that of the ID3 algorithm, and the time fluctuation was also small.

E. ANALYSIS AND SUMMARY OF EXPERIMENTAL RESULTS

Through the comparison and verification experiments, the conclusions are summarized as follows:

1) For the anomaly detection problem, the SVM without optimization ran faster, but the recognition result was somewhat different from the SVM with optimization processing. The non-optimized SVM model was suitable for recognition scenarios that required large amounts of data and required rapid training. In such scenarios, time has priority over accuracy. The two SVM models with optimization processes had strong recognition ability and could accurately identify abnormal data. These two types of models achieved good results based on indicators such as accuracy, specificity, and sensitivity. They were suitable for scenarios with high requirements for communication data security.
2) For the anomaly classification problem, on the whole, the performance of the CART model was better than the ID3 model. The CART model performed well on accuracy, achieving more than 90%. The ID3 model had an overfit phenomenon. Although it achieved good results on the training set, it did not perform well on the dataset. In the real-time Ethernet scenario of the train, the model needs to have a good ability to recognize strange data, so the ID3 model does not apply to the scenarios considered in this article. Besides, in the CART model, too many or too few nodes can affect the classification accuracy when generating a decision tree.
3) From the perspective of time complexity, the training time of the anomaly detection model with an optimization process was linearly related to the number of samples, and the running time of the attack classification model was constant with regards to the number of samples. Looking at the time, the PSO-SVM model took the longest time, followed by the GA-SVM model. The training time of SVM without optimization, ID3, and CART models were shorter.

Comprehensive experimental results, the intrusion detection of train real-time Ethernet can use the GA-SVM model for the detection of abnormal data. After passing the normal data, the CART model can be used to distinguish between the types of attacks to better complete subsequent responses and operations.

VI. CONCLUSION

In this paper, a study was conducted on two key issues of the intrusion detection problem: anomaly detection and attack classification. Designing related models were then conducted based on support vector machines and random forests algorithm in machine learning. Moreover, the introduction of particle swarm optimization and genetic algorithms for parameter optimization was done, building a train's real-time Ethernet intrusion detection experimental platform, carrying out related experiments, and verifying the model.

The achievements of this paper are as follows: Through experiments, it is proved that the PSO-SVM algorithm, GA-SVM algorithm, and CART algorithm can effectively carry out anomaly detection and attack classification in the train real-time Ethernet. From the perspective of time

complexity, it can be seen that PSO-SVM and GA-SVM anomaly model have a linear relationship, while the SVM anomaly detection model, ID3, and CART attack classification model have constant characteristics, and the time required by PSO-SVM model is higher than that of GA-SVM model, much higher than that of other models.

Train's real-time Ethernet has open interconnection characteristics, which include the continuous development of network communication technology, as well as the ever-increasing corresponding attacks and abnormal conditions. As computer technology develops, attacking methods in the continuous game of chance with intrusion detection are getting subtler. Some attack types hide the attack in application layer data, MIB library information, etc., and cannot be identified by packet and traffic characteristics. The follow-up studies may find out the impact of such covert attacks on train communication data, equipment, databases, and other aspects, and then use this as a basis to design a new attack and defense model to ensure the train's real-time Ethernet security.

REFERENCES

[1] M. Humayun, M. Niazi, N. Z. Jhanjhi, M. Alshayeb, and S. Mahmood, "Cyber security threats and vulnerabilities: A systematic mapping study," *Arabian J. Sci. Eng.*, vol. 45, no. 3, p. 19, Apr. 2020, doi: 10.1007/s13369-019-04319-2.

[2] Y. Cherdantseva, P. Burnap, A. Blyth, P. Eden, K. Jones, H. Soulsby, and K. Stoddart, "A review of cyber security risk assessment methods for SCADA systems," *Comput. Secur.*, vol. 56, no. 1, pp. 1–27, Feb. 2016, doi: 10.1016/j.cose.2015.09.009.

[3] S. Zhang, "Safety status and risk analysis of industrial control systems–one of the safety risk analyses of ICS industrial control systems," *Netw. Comput. Secur.*, vol. 1, no. 5, pp. 15–19, Jun. 2016, doi: 10.3969/j.issn.1671-0428.2012.01.006.

[4] J. F. Xue, *Intrusion Detection Technology*. Beijing, China: Posts and Telecom Press, 2016.

[5] J. P. Anderson, "Computer security threat monitoring and surveillance," James P. Anderson Company, Washington, DC, USA, Tech. Rep., 1980.

[6] D. Anderson, T. Frivold, and A. Valdes, *Next-Generation Intrusion Detection Expert System (NIDES): A Summary*. 1995.

[7] A. W. Ramanathan, "WADeS: A tool for distributed denial of service attack detection," M.S. thesis, Dept. Electron. Eng., Calgary, College Eng., Univ. Texas Austin, Austin, TX, USA, 2002.

[8] C. J. Zhou, S. Huang, N. Xiong, S. H. Yang, H. Li, Y. Qin, and X. Li, "Design and analysis of multimodel-based anomaly intrusion detection systems in industrial process automation," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 45, no. 10, pp. 2216–2468, 2017, doi: 10.1109/TSMC.2015.2415763.

[9] R. Chen, X. Li, H. Zhong, and M. Fei, "A novel online detection method of data injection attack against dynamic state estimation in smart grid," *Neurocomputing*, vol. 344, pp. 73–81, Jun. 2019, doi: 10.1016/j.neucom.2018.09.094.

[10] I. Marton, A. I. Sánchez, S. Carlos, and S. Martorell, "Application of data driven methods for condition monitoring maintenance," *Chem. Eng. Trans.*, vol. 33, no. 10, pp. 301–306, 2013.

[11] S. Adepu and A. P. Mathur, "Distributed attack detection in a water treatment plant: Method and case study," *IEEE Trans. Dependable Secure Comput.*, vol. 16, no. 1, pp. 1–14, Jan./Feb. 2018, doi: 10.1109/TDSC.2018.2875008.

[12] M. Wu and Y. B. Moon, "Intrusion detection system for cyber-manufacturing system," *J. Manuf. Sci. Eng.*, vol. 141, no. 3, pp. 7–31, Mar. 2019, doi: 10.1115/1.4042053.

[13] H. Zhao, "Research on abnormal detection algorithm of industrial control system," Metall. Automat. Res. Design Inst., Beijing, China, Tech. Rep., 2013.

[14] S. Das, A. M. Mahfouz, D. Venugopal, and S. Shiva, "DDoS intrusion detection through machine learning ensemble," in *Proc. IEEE 19th Int. Conf. Softw. Qual., Rel. Secur. Companion (QRS-C)*, Sofia, Bulgaria, Jul. 2019, pp. 471–477.

[15] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation-based anomaly detection," *ACM Trans. Knowl. Discovery Data*, vol. 6, no. 1, pp. 1–39, Mar. 2012, doi: 10.1145/2133360.2133363.

[16] I. S. Thaseen, C. A. Kumar, and A. Ahmad, "Integrated intrusion detection model using chi-square feature selection and ensemble of classifiers," *Arabian J. Sci. Eng.*, vol. 44, no. 4, pp. 3357–3368, Apr. 2019.

[17] I. S. Thaseen and C. A. Kumar, "Intrusion detection model using fusion of chi-square feature selection and multi class SVM," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 29, no. 4, pp. 462–472, Oct. 2017, doi: 10.1016/j.jksuci.2015.12.004.

[18] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, no. 5, pp. 16–27, Jan. 2016, doi: 10.1016/j.advengsoft.2016.01.008.

[19] H. Esquivel and T. Esquivel, "Router-level spam filtering using tcp fingerprints: Architecture and measurement-based evaluation," in *Proc. 6th Conf. Email Anti-Spam (CEAS)*. Mountain View, CA, USA: IEEE, 2009, pp. 1–10.

[20] J. Kim, N. Shin, S. Y. Jo, and S. Hyun Kim, "Method of intrusion detection using deep neural network," in *Proc. IEEE Int. Conf. Big Data Smart Comput. (BigComp)*, Feb. 2017, pp. 313–316.

[21] H. Y. Wang, J. H. Li, and L. Feng, "Overview of support vector mechanism and algorithm research," *Comput. Appl. Res.*, vol. 31, no. 5, pp. 1281–1286, Dec. 2014.

[22] S. H. Kok, A. Azween, and N. Jhanjhi, "Evaluation metric for crypto-ransomware detection using machine learning," *J. Inf. Secur. Appl.*, vol. 55, Dec. 2020, Art. no. 102646, doi: 10.1016/j.jisa.2020.102646.

[23] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Adv. Large Margin Classifiers*, vol. 10, no. 3, pp. 61–74, 1999.

[24] C. E. Rasmussen, *Gaussian Processes in Machine Learning* (Lecture Notes in Computer Science) vol. 3176. Feb. 2003, p. 14.

[25] Z. H. Zhou, *Machine Learning*. Beijing, China: Tsinghua Univ. Press, 2016, pp. 121–139 and 298-300.

[26] [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/

[27] A. A. Aburomman and M. B. I. Reaz, "A novel SVM-kNN-PSO ensemble method for intrusion detection system," *Appl. Soft Comput.*, vol. 38, pp. 360–372, Jan. 2016, doi: 10.1016/j.asoc.2015.10.011.

[28] W. Feng, Q. Zhang, G. Hu, and J. X. Huang, "Mining network data for intrusion detection through combining SVMs with ant colony networks," *Future Gener. Comput. Syst.*, vol. 37, pp. 127–140, Jul. 2014, doi: 10.1016/j.future.2013.06.027.

[29] I. Manzoor and N. Kumar, "A feature reduced intrusion detection system using ANN classifier," *Expert Syst. Appl.*, vol. 88, pp. 249–257, Dec. 2017, doi: 10.1016/j.eswa.2017.07.005.

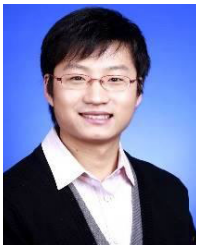
[30] P. Casas, J. Mazel, and P. Owezarski, "Unsupervised network intrusion detection systems: Detecting the unknown without knowledge," *Comput. Commun.*, vol. 35, no. 7, pp. 772–783, Apr. 2012, doi: 10.1016/j.comcom.2012.01.016.

[31] G. Qu, S. Hariri, and M. Yousif, "A new dependency and correlation analysis for features," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 9, pp. 1199–1207, Sep. 2005, doi: 10.1109/TKDE.2005.136.

RUIFENG DUO received the B.Eng. degree in electrical engineering and automation from Beijing Jiaotong University, Beijing, China, in 2019, where he is currently pursuing the master's degree in electrical engineering. His research interest includes cybersecurity of the train communication networks.

XIAOBO NIE received the B.Eng. and Ph.D. degrees from Beijing Jiaotong University, China, in 2005 and 2011, respectively. She is currently an Associate Professor with the School of Electrical Engineering, Beijing Jiaotong University. Her research interests include information physical system security and time-sensitive networks.

CHUAN YUE (Student Member, IEEE) received the B.Eng. degree in electrical engineering from Beijing Jiaotong University, Beijing, China, in 2015, where he is currently pursuing the Ph.D. degree with the Electrical Engineering School. His research interests include machine learning, network intrusion detection, and train communication network security.



NING YANG received the Ph.D. degree from the China Academy of Railway Sciences, China, in 2013. He currently works with China Academy of Railway Sciences Corporation. His research interests include control of electric traction systems and computer control networks technique.

YONGXIANG WANG received the Ph.D. degree from Beijing Jiaotong University, China, in 2009. He currently works with China Academy of Railway Sciences Corporation. His research interests include control of electric traction systems and computer control networks technique.