mathematics

MDPI

Type of the Paper (Article, Review, Communication, etc.)

Research on encryption and authentication method of industrial Automation data based on machine learning

Changsheng Ma [1], Ruchun Jia [2,*], Jing Lou[3] and Mingqian Wang [4]

[1] School of Information Engineering, Changzhou Vocational Institute of Mechatronic Technology, Changzhou, 213164,China; machangsheng@czimt.edu.cn
[2] College of Computer Science, Sichuan University, Chengdu, 610065,China;jiaruchun@stu.scu.edu.cn
[3] School of Information Engineering, Changzhou Vocational Institute of Mechatronic Technology, Changzhou, 213164,China; loujing@czimt.edu.cn
[4] School of Information Engineering, Changzhou Vocational Institute of Mechatronic Technology, Changzhou, 213164,China; wmq1989219@126.com
* Correspondence: jiaruchun@stu.scu.edu.cn; (College of Computer Science, Sichuan University)

Abstract: With the close combination of modernization and digitalization, the industrial Internet has been upgraded in an all-round way. In recent years, safety accidents of industrial control system (commonly known as industrial automation) system software have occurred from time to time. The vulnerability of the endogenous security of industrial production communication protocol is one of the key reasons for the software security accidents of industrial automation system. IDS (Intrusion Detection System) is a kind of network security technology. It can monitor the activities in the network through port scanning, network traffic analysis and so on, and identify the possible intrusion behavior. By detecting abnormal activities in the network, we can detect possible intrusion behaviors, so as to detect and prevent network security attacks in time. But the intrusion detection and Defense Technology in traditional information technology can not be applied to industrial automation system software immediately. Therefore, according to the characteristics of industrial automation system software, this paper studies the intrusion detection technology suitable for industrial automation system software, uses haqspo algorithm to improve elm and SVM, and obtains ICs intrusion detection entity model based on improved elm and ICs intrusion detection entity model based on Improved SVM. Finally, compared with QPSO, PSO and GA algorithms, the ICs intrusion detection entity model improved by haqspo algorithm has stronger main performance and can better meet the requirements of specific ICs for intrusion detection. According to the simulation results, the accuracy of ICs intrusion detection model composed of stacked classifiers is higher than that of single classifier model, while the false negative rate and false negative rate are lower than that of single classifier model.

Keywords: Machine learning; Industrial automation; Safety data; Encryption authentication

MSC:

Citation: To be added by editorial staff during production.

Academic Editor: Firstname Lastname

Received: date
Revised: date
Accepted: date
Published: date

1. Introduction

In order to cope with the increasingly severe ICS network threat situation, the traditional IT network security technology can not be immediately applied to industrial production scenarios, and there is an urgent need to formulate a reasonable security protection programme for industrial production scenarios. At this stage, the key to ICS network security technology is intrusion detection technology, intrusion detection technology and data encryption [1]. Intrusion detection is often regarded as the second security gate after the server firewall. It is an effective filler to the server firewall and is a

virus protection technique [2]. It is based on the collection of system software and online information content, and the analysis and development of that information content, which in turn assists the system software in detecting network intrusions and responding instantly [3]. Intrusion detection technology can not only test external attacks, the system can also effectively test internal attacks, is a set of security protection, detection, response in one of the virus protection technology [4]. In this paper, device learning method is used to construct intrusion detection entity model, which provides an effective solution for the security protection of ICS Internet.

In this paper, the ELM algorithm is used in the application domain to practice the ICS intrusion detection data information after rough set feature approximation to get the ICS intrusion detection entity model based on ELM, and then the entity model is detected and the various performance parameters of the model are analysed.

1.1. Extreme Learning Machine

Traditional feedforward control neuron networks (e.g., BP) are used in many industries, but some of their original drawbacks limit their development trend. Feedforward control neuron networks mostly use gradient descent, but this method has the following three drawbacks:

1. In order to better calibrate the weights and thresholds, gradient descent usually goes through several iterations, and reaching the goal will lead to too much practice time;
2. The part that is easy to sink is the best;
3. Most of the feedforward control neuron networks are characterised by the appropriateness of the choice of the learning rate, which is the only way to get good performance [5].

If the learning rate is too large, it is likely to cause non-convergence in practice. If the learning rate is too small, the convergence will be slower. In order to better solve the shortcomings of the traditional feedforward control neuron network, Guo Juanjuan et al [6] people based on the theory of theoretical inverse matrix theory, through the Internet explicitly proposed a new single hidden layer feedforward control - ELM. the algorithm gives an arbitrary connection, the input weights of the keying layer and the hidden layer, as well as the thresholds of the nodes of the hidden layer in the subsequent exercises of the whole process do not need to be adjusted [7]. The only basic parameters that must be set by the algorithm are the hidden layer neurons, the optimal solution for the Internet can be obtained in one step, and the ELM has a stronger generalisation ability and faster learning and training than traditional neural networks using gradient descent (e.g. BP).

A single hidden layer feedforward control neuron network consists of three parts, i.e., the typing layer, the hidden layer and the derived layer. Let the number of neuronal cells in the Internet typing layer be n, the number of neuronal cells in the hidden layer be m, and the number of neuronal cells in the derived layer be w. There are Q different training samples. The training samples are divided into input drainage matrix X and derived drainage matrix Y. The input drainage matrix X is:

$$X = \begin{bmatrix} \chi_{11} & \chi_{12} & \cdots & \chi_{1Q} \\ \chi_{21} & \chi_{22} & \cdots & \chi_{2Q} \\ \vdots & \vdots & \vdots & \vdots \\ \chi_{n1} & \chi_{n2} & \cdots & \chi_{nQ} \end{bmatrix}_{nxQ} \quad (1)$$

The output matrix Y is:

$$Y = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1Q} \\ y_{21} & y_{22} & \cdots & y_{2Q} \\ \vdots & \vdots & \vdots & \vdots \\ y_{m1} & y_{m2} & \cdots & y_{mQ} \end{bmatrix}_{mxQ} \tag{2}$$

The output T of the network is given by Fig. 1:

$$T = \left[t_1, t_2, \cdots t_Q\right]_{mxQ}, tj = \begin{bmatrix} t_{1j} \\ t_{2j} \\ \vdots \\ t_{mj} \end{bmatrix}_{m\times 1} = \begin{bmatrix} \sum_{i=1}^{1} \beta_{i1} g\left(\varpi_i x_j + b_i\right) \\ \sum_{i=1}^{1} \beta_{i2} g\left(\varpi_i x_j + b_i\right) \\ \vdots \\ \sum_{i=1}^{1} \beta_{im} g\left(\varpi_i x_j + b_i\right) \end{bmatrix}_{m\times 1} \quad \left(j = 1,2,\cdots Q\right) \tag{3}$$

where g(·) is the activation function, $b = \left[b_1, b_2, \cdots b_i\right]_{i\times 1}^{T}$ is the threshold of the hidden layer node, ϖ is the input weight connecting the key-in layer to the hidden layer and β is the derived weight connecting the hidden layer to the input - output layer, And the actual way ϖ and β are shown below.

The output weights ϖ can be expressed as:

$$\varpi = \begin{bmatrix} \varpi_{11} & \varpi_{12} & \cdots & \varpi_{1n} \\ \varpi_{21} & \varpi_{22} & \cdots & \varpi_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ \varpi_{l1} & \varpi_{l2} & \cdots & \varpi_{ln} \end{bmatrix}_{lxQ} \tag{4}$$

The output weights β can be expressed as:

$$\beta = \begin{bmatrix} \beta_{11} & \beta_{12} & \cdots & \beta_{1m} \\ \beta_{21} & \beta_{22} & \cdots & \beta_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ \beta_{l1} & \beta_{l2} & \cdots & \beta_{lm} \end{bmatrix}_{lxm} \tag{5}$$

The output T of the network can also be expressed in the following equation:

$$H\beta = T' \tag{6}$$

Where H is the output matrix of the implicit layer of the network and T' is the transpose of the output T of the network, the specific form of H is as follows.

The output matrix H of the implicit layer is:

$$H = \begin{bmatrix} g\left(\varpi_1 \cdot x_1 + b_1\right) & g\left(\varpi_2 \cdot x_1 + b_2\right) & \cdots & g\left(\varpi_l \cdot x_1 + b_l\right) \\ g\left(\varpi_1 \cdot x_2 + b_1\right) & g\left(\varpi_2 \cdot x_2 + b_2\right) & \cdots & g\left(\varpi_l \cdot x_2 + b_l\right) \\ \vdots & \vdots & \vdots & \vdots \\ g\left(\varpi_1 \cdot x_Q + b_1\right) & g\left(\varpi_2 \cdot x_Q + b_2\right) & \cdots & g\left(\varpi_l \cdot x_Q + b_l\right) \end{bmatrix}_{Qxl} \tag{7}$$

When there exists an infinitely differentiable activation function g(·) in a random interval and the number of neurons in the hidden layer is equal to the number of samples in the training set, a random key-in weight and a hidden layer node Threshold

b, the feedforward control neuron network can approximate these Q training samples with zero bias, i.e:

$$\sum_{j=1}^{Q}\left\|t_j - y_j\right\| = 0 \tag{8}$$

$$\sum_{j=1}^{Q}\left\|t_j - y_j\right\| = 0 \tag{9}$$

In specific applications, the number of samples in the training set is usually very large, and the number of hidden layer neurons is usually lower than the number of samples in the training set, and an increase in the number of hidden layer neurons may prompt computation [8]. When an approximation of 0 deviation cannot be accomplished, an arbitrarily small deviation is given $\varepsilon > 0$, The training deviation of the feedforward control neuron network can be approximated as the error ε , i.e:

$$\hat{\beta} = H^{+}T, \tag{10}$$

where H^{+} is the Moore-penrose generalised inverse of H.

In general, the ELM algorithm key has the following four processes:

1. Set the main parameters of ELM. Determine the number of neurons in the hidden layer, and arbitrarily give the input weights ϖ and thresholds b of the hidden layer nodes.
2. Select the activation functions for infinite differentiation, such as the S-function formula, sinusoidal function, and composite function.
3. compute the derived drainage matrix H of the hidden layer from (Equation 7).
4. compute the derived weights β from (Equation 10).

1.2. Extreme Learning Machine Intrusion Detection Model for Industrial Control Systems

The modelling of IDS was first explicitly proposed by Denning in 1987, but with the rapid development of IDS, many new IDS will no longer fully conform to the entity model, and IDS is too purposeful to be generalised [9]. The inter-application of systems in turn limits the development trend of IDS [10]. Therefore, Chen et al. explicitly proposed the CIDF entity model as shown in Figure 1 below:

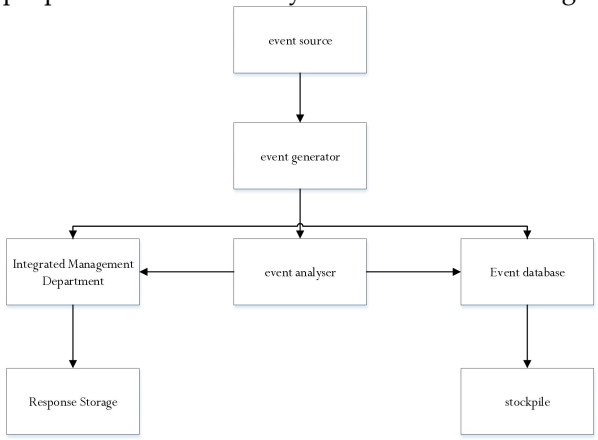

Figure 1. Generic Intrusion Detection Framework.

In Fig. 1, the key function of the event generator is the collection of information content, including the content of information about the Internet, the client and the operating system, as well as their personal behaviours, suitable for providing data and

information in the subsequent solution [11]. The event parser is the key to the whole intrusion detection architecture and its main function is to analyse the collected data and check for any unusual individual behaviour and the main function of the response module is to respond to the results obtained from the event parser analysis [12]. If signs of intrusion are detected, it issues an alert and takes some mandatory personal actions such as disconnecting, changing file permissions, etc [13]. The event database query function stores the collected events for the intrusion detection system software to use when needed [14]. The main objective of the article is to conduct a scientific study on the event parser to make it clear and specific, and to explicitly propose an ELM-based entity model for ICS intrusion detection, including the industrial production data preprocessing control module, the ELM motion control module and its ELM detection control module. It is shown in Figure 2 below.

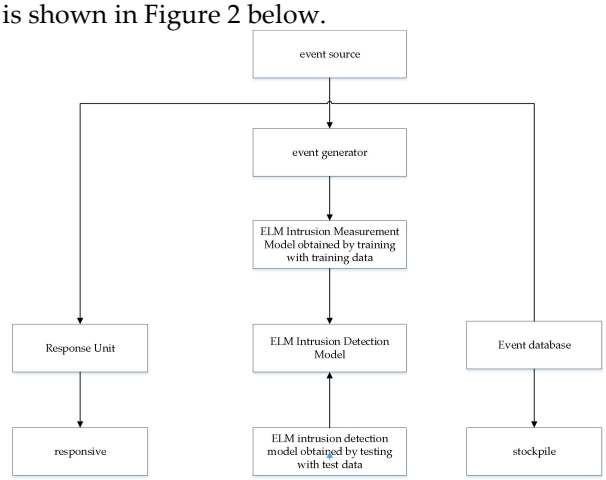

Figure 2. ELM-based Intrusion Detection Model for ICS.

1. Industrial production data preprocessing control module: the collected industrial production data information is preprocessed, including standardisation, normalisation and feature approximation of template identification, and then the preprocessed data information is divided into practice data information and data testing, which provides data information in accordance with the file format for the subsequent work.
2. ELM exercise control module: according to the ELM algorithm for the exercise data information after the completion of the preparation process, according to the ELM to get the ICS intrusion detection entity model, which can distinguish between all normal personal behaviour and abnormal personal behaviour ICS in the behaviour. The purpose of intrusion detection [15].
3. ELM detection and control module: the preprocessed data test is sent to the control module, according to the motion control module to obtain the ELMICS intrusion detection entity model, the data test is detected and the validity of the entity model is verified. In specific applications, the data information to be detected can be sent to the control module, and the intrusion detection entity model will be detected according to the detected data information to identify the presence of intrusive personal behaviour.

2. Improvement of machine learning based intrusion detection algorithms for industrial control systems

As mentioned above, the number of hidden layer neurons in ELM has been clarified through several laboratories, and the inputs of weights and thresholds of hidden layer nodes are likely to cause generalisation over the Internet. Therefore, an algorithm must be used to optimise the two main parameters, the key weights and the thresholds of the hidden layer nodes. For SVMs with a Gaussian kernel function, the classification feature penalises the appropriateness of the chosen values of the main parameter c and the kernel function parameter g. The main parameters c and g are specified using the grid search method, but the grid search method must analyse any point in the xml grid graph

for a longer period of time until the best main parameters are found [16]. Therefore, an algorithm must be used to optimise the main parameters c and g to reduce the training time. The paper explicitly proposes the HAQSPO algorithm to deal with the problem of improving the main parameters and then the best main parameters will be found to construct the intrusion detection entity model.

2.1. Quantum particle swarm optimisation algorithm QPSO algorithm

Jun Sun et al. explicitly proposed that the particles of this algorithm are characterised by quantum technology individual behaviour compared to the PSO algorithm. The algorithm uses a particle in quantum space to represent each individual and usually uses velocity and position to describe the particle. However, it is not possible to figure out both the efficiency and the position of the particle in quantum space [17]. In order to better describe a particle in quantum space, we choose to apply the wave function V to the description and then choose the Schrödinger equation to elucidate the particle. At this point, the Mmonte Carlo method (inverse transformation method) can be used to obtain the particle position upgrade formula. The formula is as follows:

$$x_{id}(t+1) = p_i(t) \pm a(t)\left|C_d(t) - x_{id}(t)\right| \times \ln\left[1/\mu_{id}(t)\right] \tag{11}$$

where xid (t+1) is the position of particle i in the d-dimension at the t+1st iteration; ± is determined by the magnitude of μ, which is a random number co-distributed in the middle of (0, 1), and takes a negative sign when μ > 0.5, and takes a negative sign in all other cases; a is called the augmentation index, and it is the only performance indicator other than the size of the crowd management and the frequency of iteration; and pi (t) is an arbitrary portion of a particle at the tth iteration, the The calculation formula is as follows:

$$p_i(t) = \varphi_d(t) p_{id}(t) + \left[1 - \varphi_d(t)\right] p_{gd}(t) \tag{12}$$

In Eq. pid is the best position of the individual and pgd is the best position of the population. φ_d is the same value of μ chosen in Eq. 11, which are random numbers co-distributed in the middle of (0, 1). In Eq. (Eq. 11), C(t) is the best position of the mean, which is calculated as follows:

$$C(t) = \frac{1}{N} \sum_{i=1}^{N} p_i(t) \tag{13}$$

2.2. Hybrid adaptive quantum particle swarm optimisation algorithm

In the middle and late iterations, the QPSO algorithm suffers from the problems of mature convergence and deep trapping due to low species diversity. To address this problem, the article explicitly proposes the HAQSPO algorithm, which includes the following three improvements: firstly, differential signalling countermeasures are applied to improve the release of any part of the particle, then responsive change control measures are used to manipulate the expansion exponent a, and finally, Levy sailing countermeasures are added to the upgrade of the particle part, which exploits the properties of Levy sailing countermeasures to improve the performance of the species [18].

2.2.1. Improvements in particle random position updating

The key feature of the DE algorithm is differential signalling confrontation. In the article, the DE/rand/1/bin differential signal confrontation is closely combined with the upgrade formula calculation of particle arbitrary part for improvement. The improved formula is calculated as follows:

$$p_i(t+1)=\varphi_d(t)\times p_{(r0)d}(t)+[1-\varphi_d(t)] \times F\times\left(p_{(r1)d}(t)-p_{(r2)d}(t)\right) \tag{14}$$

In the formula, $p_{(r0)d}(t)$, $p_{(r1)d}(t)$, and $p_{(r2)d}(t)$ are three different personal best positions chosen at random. Corpse is a factor for resizing pictures, inspired by the responsive weight value PSO algorithm. For the operation of the picture size adjustment factor, the main parameter corpse response is used in the article. Changing the formula in such a way that the formula is calculated as follows:

$$F=\begin{cases} F_{\min}-\dfrac{(F_{\max}-F_{\min})\times(f-f_{\min})}{f_{avg}-f_{\min}}; f\le f_{avg} \\ F_{\max}; f>f_{avg} \end{cases} \tag{15}$$

According to the references, the maximum and minimum values of the image size factor F in the DE algorithm are usually 0.9 and 0.4. f is the optimal individual fitness value, and favg and fmin denote the average fitness value and the minimum fitness value of the population, respectively. Since the main parameter F is adjusted with the change of particle fitness value, it is called the response of the main parameter.

2.2.2. Improvements in parameter a control methods

Changes in the main parameter a in Equation (2-1) immediately impair the individual behaviour of the particle. The manipulation of the main parameter a is usually chosen as a fixed value and linear normalisation method. Same as the manipulation of image size factor F, in this paper, the main parameter a is adjusted by using the responsive change control measure, which is calculated as follows:

$$a=\begin{cases} a_{\min}-\dfrac{(a_{\max}-a_{\min})\times(f-f_{\min})}{f_{avg}-f_{\min}}; f\le f_{avg} \\ a_{\max}; f>f_{avg} \end{cases} \tag{16}$$

The maximum and maximum values of amax and amin are taken in the equation respectively. Based on the scientific study of the control measures of the main parameter a in the references, it is found that the maximum and minimum values of a are chosen to be 1 and 0.5, respectively. the practical results are good.

2.2.3. Improvements in particle position update

Levy's navigation strategy is ideal for modelling the search for food by small animals in unfamiliar natural environments. It is a non-Gaussian stochastic process that combines frequent short-circuit partial retrieval and sometimes long-range global search [19]. In view of the characteristics of Levy nautical confrontation, many experts and scholars have added Levy nautical confrontation to the evolution equations of bionic intelligence algorithms, and achieved better practical results. Applying the Levy navigation path relation equation explicitly proposed by Mantegna:

$$Levy(\lambda)=\mu/|v|^{1/\beta} \tag{17}$$

where the main parameter β is usually β = 1.5; the parameters μ and v are $\mu\sim N(0,\ \sigma_{\mu}^{2})$, $v\sim N(0,\ \sigma_{v}^{2})$, respectively; and the standard deviations σ_{μ} and σ_{v} of the normal distributions corresponding to the main parameters u and v are taken to satisfy Eq. (Eq. 18):

$$\begin{cases} \sigma_\mu = \left\{ \dfrac{\Gamma(1+\beta)\times \sin(\pi\beta/2)}{\Gamma[(1+\beta)/2]\times\beta\times 2^{(\beta-1)/2}} \right\} \\ \sigma_v = 1 \end{cases} \tag{18}$$

Substituting Eq. (Calculation 18) into Eq. (Calculation 17) yields the Levy sailing route. After adding the Levy sailing confrontation to the QPSO algorithm, the particle evolution equation becomes:

$$\begin{cases} x_{id}(t+1) = Levy(\lambda)[x_{id}(t) - gbest_{id}] + p_{id}(t) + \\ \alpha(t)|C_d(t) - x_{id}(t)| \times \ln[1/u_{id}(t)], u_{id}(t) > 0.5 \\ x_{id}(t+1) = Levy(\lambda)[x_{id}(t) - gbest_{id}] + p_{id}(t) - \\ \alpha(t)|C_d(t) - x_{id}(t)| \times \ln[1/u_{id}(t)], u_{id}(t) \le 0.5 \end{cases} \tag{19}$$

Where gbestid is the fraction of particles with the best fitness value in the population. The following is the implementation process of HAQSPO algorithm: Step1 Reset the number of particle swarms, larger iteration frequency, and retrieve the position of each particle in the particle swarm in the indoor space and reset the optimal position of individual particles to the optimal position of today; Step2: Calculate the optimal position of the average of particle swarms by applying the formula (Eq. 13); Step3 Calculate the fitness value of the particles and calculate the global optimal position of the upgraded particles and global optimal position of the upgraded particles according to the minimum fitness criterion to calculate the individual optimal position of the upgraded particle and the global optimal position of the species; Step4: Apply the formula (Eq. 14) to measure the position of any point; Step5: Apply the formula (Eq. 19) to measure the new portion of the particle; Step6: Repeat Step2~Step5 until the set completion criterion is reached or a greater iteration frequency is reached.

2.3. Intrusion Detection Algorithm for Industrial Control Systems Based on Optimised Extreme Learning Machine

The hidden layer derived drainage matrix of the ELM is calculated based on the key-in weight values and thresholds of the hidden layer nodes. When these are pressed arbitrarily, some non-optimal key-in weight values and hidden layer nodes are likely to appear. The threshold value leads to the classification of ELM is prone to problems such as generalisation ability and suboptimal accuracy [20]. In order to better deal with this problem, the article chooses the HAQSPO algorithm to improve the ELM-based ICS intrusion detection model, i.e., the HAQSPO algorithm is chosen to find the optimal key-in weight values and the thresholds of the hidden layer nodes of the ELM, so as to calculate the resulting weight values. The steps to construct the NRS-HAQSPO-ELM intrusion detection model are as follows:

Step1: Normalise the data and divide it into training and detection sets; Step2: Reset the number of particle swarms, the larger iteration frequency, and the position of each particle in the particle swarm, and reset the optimal position of the particle to Location; Step3: Apply the formula (Eq. 13) to compute the optimal position of the particle swarm mean; Step4: Take the ELM key-in weights value and the threshold of the hidden layer node as the improvement objective, and the fitness is opposite to the classification accuracy; Step5: Apply HAQSPO algorithm for iterative optimisation according to the minimum fitness rule to retrieve the individual optimal solution and the global optimal solution. Step6: Measure the specific position of any point and the new part of the particle; Step7: Repeat Step2~Step5 until the set criterion is reached or a greater iteration frequency is reached. Step8: Apply the searched optimal key-in weights and the

thresholds of hidden layer nodes to construct the ELM support vector machine model, and finally get the ICS intrusion detection model based on NRS-HAQSPO-ELM.

2.4. Intrusion Detection Algorithm for Industrial Control Systems Based on Optimised Support Vector Machines

The flow of ICS intrusion detection model construction based on NRS-HAQSPO-SVM is shown in Fig. 3.

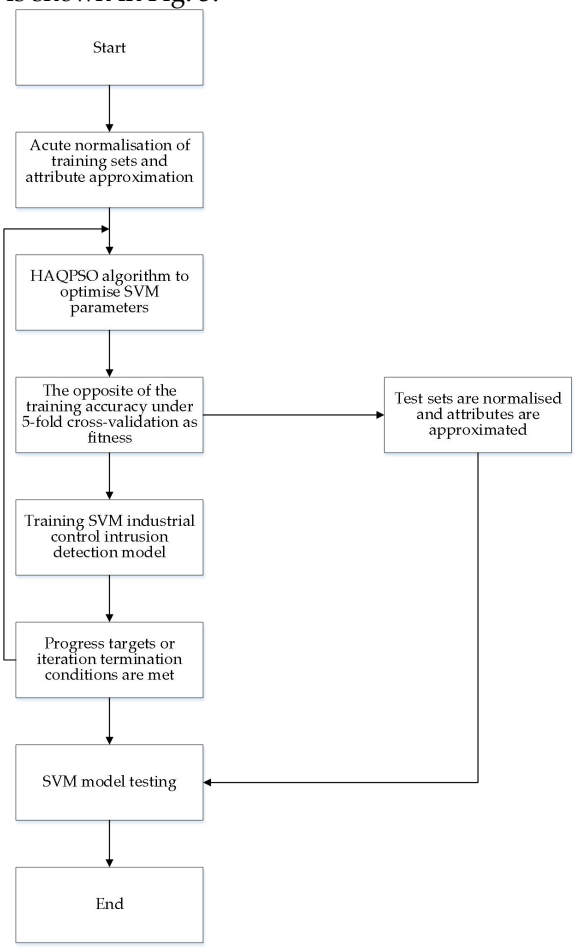

Figure 3. Flowchart of ICS intrusion detection construction based on NRS-HAQSPO-SVM.

The whole model construction process is simply divided into three steps:

1. Preparation and processing of data information: the data are typed with sample patterns, divided into training set and detection set, and normalised respectively, and then the field Rough set is applied for feature approximation;
2. Optimisation of the main parameters of SVM is explicitly defined: the main parameters c and g of SVM are taken as the target of improvement, and classification accuracy under the 5-fold cross-validation is selected The opposite number as the fitness, use HAQSPO algorithm to find the best SVM main according to the iterative parameters, use the training set to practice and get the SVM model.
3. Model testing: create a model with the best main parameters found and then test the model using the preprocessed testing set.

3. Analysis of experimental results

In order to better verify the practical effect of HAQPSO algorithm, the algorithm proposed in this paper and QPSO, PSO, GA algorithms to improve the ELM and SVM, respectively, before and after the construction of data based on ICS intrusion reduces the

respective application characteristics of the improvement of the detection model and basis of the ELM. Improve the ICS intrusion detection model of SVM and compare and analyse the test results. The main parameters that must be set for the HAQPSO algorithm are: the contraction expansion index cr is [0.5, 1], the range of values of the image size factor F is [0.4, 0.9], and the main parameter of Levy navigation is 1.5.The main parameters that are set for the HAQPSO algorithm and the other algorithms in the article are: a larger number of iterations of 50, and a crowd management size of 20.The search category for SVM is [0.001, 1000]. The number of hidden layer neurons in ELM is 500 and the search category is [-1,1]. The retrieval dimension d is determined by the number of hidden layer neurons and the number of typed layer neurons. The calculation method is as follows:

$$d = NumberofHiddenNeurons * \left(NumberofHiddenNeurons + 1\right) \quad (20)$$

The HAQPSO algorithm has the highest accuracy in optimising the ELM with 98.54%, while the GA algorithm has the lowest accuracy with 97.88%. Regarding the convergence speed of the algorithms, the QPSO algorithm converges faster and can converge to the optimal value in the 15th generation; followed by the HAQPSO algorithm, which can converge to the optimal value in the 19th generation; and the GA algorithm converges slower. The different algorithms improve the fitness value of the whole process of SVM training by using the training set with reduced features to train the model [21].

The HAQPSO algorithm has the highest optimisation accuracy of 99.11% for SVM, while the GA algorithm has the lowest accuracy of 98.25%. In terms of the convergence speed of the algorithms, the QPSO algorithm converged faster, with the best convergence in the 20th generation; the HAQPSO algorithm was the next best, with the best convergence in the 25th generation; and the GA algorithm had the best convergence. Compared with the unenhanced NRS-ELM model, the practical results of the NRS-HAQPSO-ELM model are improved to a certain extent, especially for the three types of data information of MSCI, MFCI and Dos, and the total area under the load of ROC curves is greatly improved from 0.73 to 0.94, 0.87 to 1, and 0.72 to 0.81, respectively, which indicates that the HAQPSO algorithm has a good effect on the improvement of NRS-ELM model and can find the appropriate key-in weights and thresholds of hidden layer nodes to improve the practical effect of detection [22-23]. Compared with the unenhanced NRS-SVM model, the practical effect of the NRS-HAQPSO-SVM model is improved to a certain extent, especially for the three types of data information, MSCI, MFCI, and Dos, the total area under the ROC curve network is greatly improved from 0.82 to 0.94, from 0.88 to 1, and from 0.76 to 0.92, respectively and the HAQPSO algorithm improves the NRS-ELM model as well as the practical effect of HAQPSO The optimisation algorithm of the NRS-SVM model is good enough to find the appropriate main parameters c and g to improve the practical effect of the test [24]. In order to show the advantages of HAQPSO algorithm better and faster, the algorithm proposed in this paper is upgraded with QPSO, PSO and GA algorithms to ELM and SVM, respectively, with different data structures after applying feature reduction. The algorithm is optimised according to the ICS of ELM. The intrusion detection model and different boosting algorithms are optimised according to the SVM ICS intrusion detection model and simulation experiments are carried out. The results of performance parameters are shown in Fig. 4.

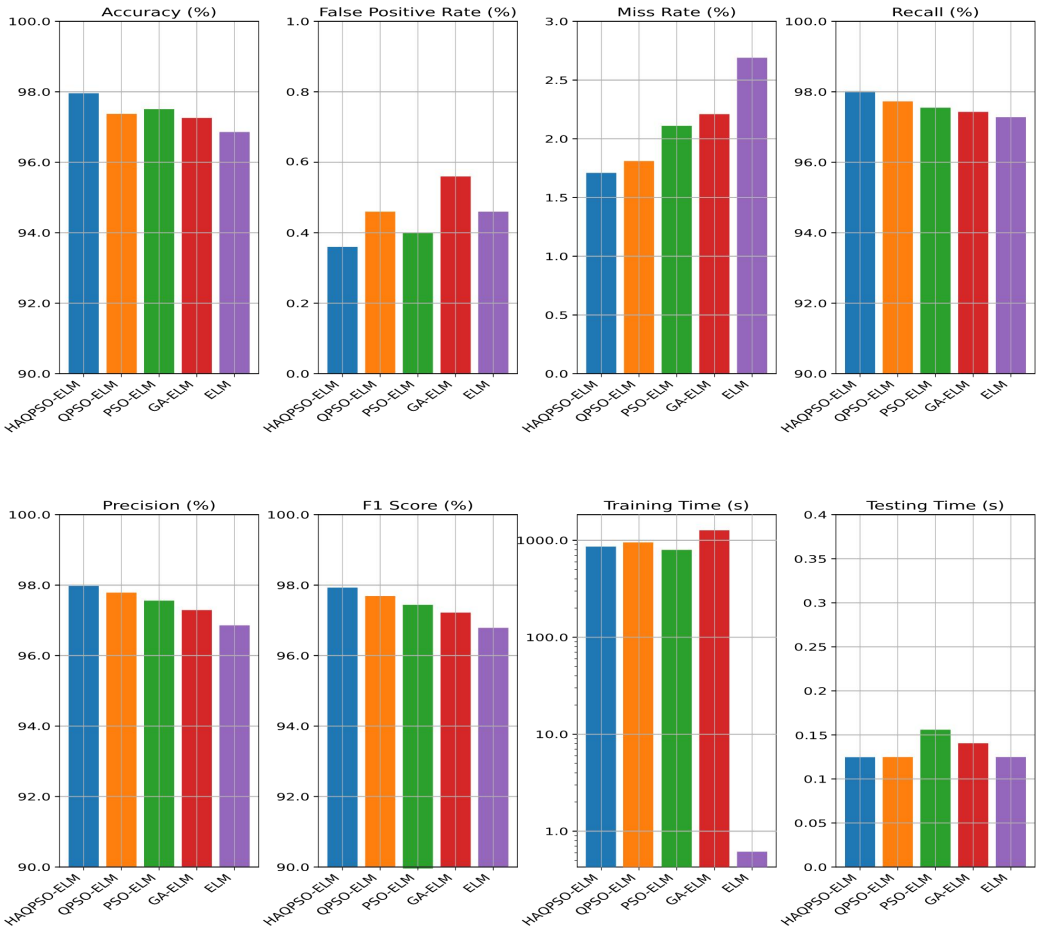

Figure 4. Comparison of performance metrics of ELM optimised by different optimisation algorithms.

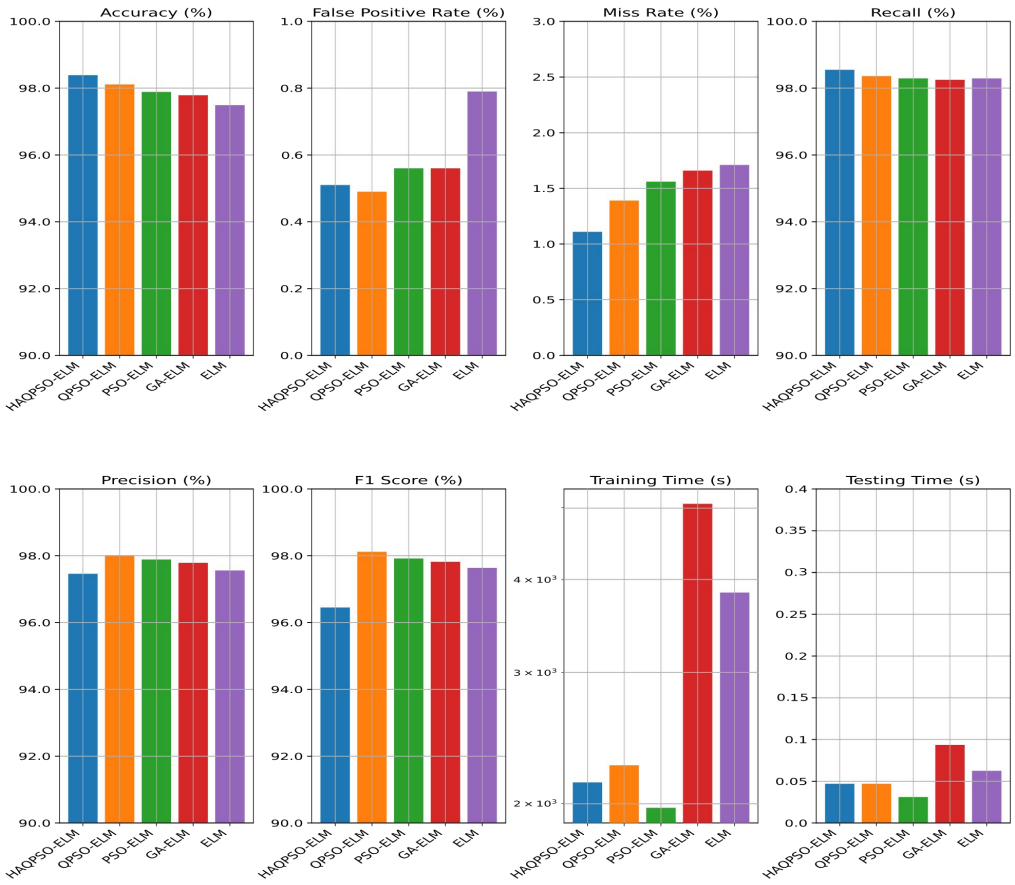

Figure 5. Comparison of performance metrics of SVMs optimised by different optimisation algorithms.

From Fig.4, it can be seen that compared to the unoptimised ELM based ICS Intrusion Detection Model, the optimised model of the optimisation algorithm has some improvement in detection accuracy, but the optimisation algorithm has to take a certain amount of time before it can enter the WEIGHT VALUE Optimised thresholds for the hidden layer nodes, thus increasing the practice time. In terms of accuracy, precision, recall and F1 metric values, HAQPSO algorithm is the largest, reaching 97.95%, 98.03%, 97.99% to 97.92%, KW 97.25%, 97.42%, 97.3% and 97.21%, respectively. ; In terms of leakage rate and miss rate, the HAQPSO algorithm has only 0.5% and 1.1%, which is the least among all the algorithms. It indicates that compared with any given main parameters, the optimisation algorithm can not only find more suitable main parameters, but also improve the self-learning ability and generalisation ability of the model to a certain extent, and reduce the leakage rate and underreporting rate [25]. Among several optimisation algorithms, the HAQPSO algorithm has the best main performance. In terms of practice time and detection time, the HAQPSO algorithm consumes very less time and is only higher than the PSO algorithm. The results show that the HAQPSO algorithm is able to meet the efficient requirements for intrusion detection in specific industrial production scenarios to a certain extent compared to the QPSO, PSO and GA algorithms. As can be seen from Fig. 5, the optimised algorithm is based on the SVM ICS intrusion detection model. RESULTS AND OPTIMISATION Based on the ELM ICS intrusion detection model, the HAQPSO algorithm is significantly better than the other optimisation algorithms, and is only higher than the PSO algorithm at the level of practice time and detection time, which are only 2139.6629s and 0.0468 seconds, which is much smaller than the 3845.4461s of the grid search method at the level of practice time. However, the time required for the optimisation of the SVM-based ICS intrusion detection model also takes at least 1974.099s, which is still time-consuming compared to

the ELM-based ICS intrusion detection model [26]. The actual results of each optimisation algorithm are analysed based on the actual results of the data information of the eight attacks. Below is an example of optimising the ICS intrusion detection model based on SVM.The accuracy of NRS-HAQPSO-SVM for each method of attack detection is usually the greatest, especially in detecting MSCI and Dos attacks, which is significantly higher than the SVM intrusion detection model optimised by other algorithms.

According to the data analysis of the above simulation experiment results, the HAQPSO algorithm is optimised based on the ELM ICS intrusion detection model, which greatly improves the actual detection effect; the HAQPSO algorithm is optimised based on the SVM ICS intrusion detection model, which not only reduces the practicing time, but also improves to a certain extent, compared to the SVM applying the network grid search method to retrieve main parameters the degree of practical effectiveness of the inspection.

4. Conclusions

In the field of machine learning, it is found that the classification accuracy of composition classifiers is usually higher than that of individual classifiers. In order to better improve the main performance characteristics of ICS intrusion detection models, the article explicitly proposes an ICS intrusion detection algorithm based on Stacking compositional classifiers. Optimised ELM, optimised SVM and XGBoost are first used as the base learning and training algorithm, followed by regression analysis algorithm. For the meta-learning algorithm, an ICS intrusion detection model is built which constitutes a Stacking-based classifier. Based on the comparison of simulation experiment results, the accuracy of the ICS intrusion detection model consisting of stacking classifiers is higher than that of the single classifier model, and the false-negative rate and the under-reporting rate are lower than that of the single classifier model.

References

1. Li Pan, Zhao WT, Liu Q, et al. Security Issues and Their Countermeasuring Techniques of Machine Learning: A Survey[J]. *Journal of Frontiers of Computer Science & Technology*, 2018; Volume 12, pp. 171-184.
2. B Chatterjee, D Das, S Maity, et al. RF-PUF: Enhancing IoT Security Through Authentication of Wireless Nodes Using In-Situ Machine Learning[J]. *IEEE Internet of Things Journal*, 2018; Volume 6, pp. 388-398.
3. Jia Chunfu, Wang Yafei, Chen Yang, et al. Machine learning algorithm for a homomorphic encrypted data set[J]. *Journal of Tsinghua University(Science and Technology)*, 2020; Volume 60, pp. 456-463.
4. Zhou W, Zhang SK, Ding Y, et al. Adversarial Example Attack Analysis of Low-Dimensional Industrial Control Network System Dataset[J]. *Journal of Computer Research and Development*, 2020; Volume 57, pp. 736-745.
5. Sun Xin, Fang Fang, Sun Chang Hua. Research on multi?dimensional industrial system security perception method based on machine learning and specific attack characteristics[J]. *electronic design engineering*, 2019; Volume 27, pp. 117-121, 126.
6. Guo Juanjuan, Wang Qiongxiao, Xu Xin, et al. Secure Multiparty Computation and Application in Machine Learning[J]. *Journal of Computer Research and Development*, 2021; Volume 58, pp. 2163-2186.
7. Wu Li-Hao, Zhang Xiao-Shu, Liang Xue-Feng. Research on Automatic Identification and Security Risk of System Login Verification Code based on Machine Learning[J]. *Chinese Journal of Health Informatics and Management*, 2020; Volume 17, pp. 523-527.
8. Liu K, Ma SH, Ma O, et al. Secure Control for Cyber-physical Systems Based on Machine Learning[J]. *Acta Automatica Sinica*, 2021; Volume 47, pp. 1273-1283.
9. Qiu Lingfeng, Hu Xiaofeng, Zhou Rui, et al. Research on the occurrence regularity of typical social security incidents based on machine learning and implications for Xiongan New Area[J]. *Journal of Safety Science and Technology*, 2018; Volume 14, pp .11-17.
10. Chang Zhengwei, Peng Qian, Chen Ying. Safety Supervision Method for Power Operation Site Based on Machine Learning and Image Recognition[J]. *Electric Power*, 2020; Volume 53, pp. 155-160.
11. Li Pan, Zhao WT, Liu Q, et al. Security Issues and Their Countermeasuring Techniques of Machine Learning: A Survey[J]. *Journal of Frontiers of Computer Science and Technology*, 2018; Volume 12, pp. 171-184.
12. Zhong Sheng, Zhong Fuli, Xie Yuhao, et al. Design of Automatic Control System for Safe Operation of Key Sections of Power Grid Based on Machine Learning[J]. *Techniques of Automation and Applications*, 2021; Volume 40, pp. 48-52, 61.

13. Zhao Jiye. Network space mimicry security hierarchical detection technology based on machine learning algorithm[J]. *electronic design engineering*, 2021; Volume 29, pp. 121-125.
14. Xia Yuan, Liu Dongfeng, Zhang Jinkui, et al. BERT-based automated risk of bias assessment[J]. *Chinese Journal of Evidence-Based Medicine*, 2021; Volume 21, pp. 204-209.
15. Li XJ, Wu GW, Yao L, et al. Progress and Future Challenges of Security Attacks and Defense Mechanisms in Machine Learning[J]. *Journal of Software*, 2021; Volume 32, pp. 406-423.
16. Yu YC, Ding L, Chen ZN. Research on Attacks and Defenses towards Machine Learning Systems[J]. *Netinfo Security*, 2018; pp. 10-18.
17. Chatterjee B, Das D, Sen S. RF-PUF: Enhancing IoT Security through Authentication of Wireless Nodes using In-situ Machine Learning[J]. *IEEE Internet of Things Journal*, 2018; Volume 6, pp. 388-398.
18. Mawgoud A A, Karadawy A I, Tawfik B S. A Secure Authentication Technique in Internet of Medical Things through Machine Learning[J]. *arXiv preprint arXiv:1912.12143*, 2019.
19. Chaitanya G K, Sekhar K R. Verification of pattern unlock and gait behavioural authentication through a machine learning approach[J]. *International Journal of Intelligent Unmanned Systems*, 2022; Volume 10, pp. 48-64.
20. Gupta B B, Prajapati V, Nedjah N, et al. Machine learning and smart card based two-factor authentication scheme for preserving anonymity in telecare medical information system (TMIS)[J].*Neural Computing and Applications*, 2021; Volume 35, pp. 5055-5080.
21. Nyakomitta P S, Nyangaresi V O, Ogara S O. Efficient Authentication Algorithm for Secure Remote Access in Wireless Sensor Networks[J].*Journal of Computer Science Research*, 2021; Volume 3, pp. 43-50.
22. Borkar G M, Mahajan A R. Security aware dual authentication-based routing scheme using fuzzy logic with secure data dissemination for mobile ad-hoc networks[J].*International Journal of Communication Networks & Distributed Systems*, 2018; Volume 21, pp. 157-186.
23. Praseetha V M, Bayezeed S, Vadivel S. Secure Fingerprint Authentication Using Deep Learning and Minutiae Verification[J].*Journal of Intelligent Systems*, 2019; Volume 29, pp. 1379-1387.
24. El-Hajj M, Fadlallah A, Chamoun M, et al. Secure PUF: Physically Unclonable Function based on Arbiter with Enhanced Resistance against Machine Learning (ML) Attacks[C], *SEIA'2019 Conference Proceedings*, 2019; pp. 216.
25. Kaushal S, Buksh B. A study secure multi authentication based data classification model in cloud based system[J]. *International Journal of Advances in Applied Sciences*, 2020; Volume 9, pp. 240.
26. Lee K, Esposito C, Lee S Y. Vulnerability Analysis Challenges of the Mouse Data based on Machine Learning for Image-based User Authentication[J]. *IEEE Access*, 2019; Volume 7, pp.177241-177253.