



# OPEN Network security analysis based on feature selection and optimized fireworks algorithm

Liang Zhou, Chang Liu, Li Tian, Jie Wang, Chang Liu & Xiao Yu✉

Traditional network security analysis methods exhibit critical limitations in processing high-dimensional dynamic data, including inefficient feature selection, poor adaptability to evolving threats, and low detection sensitivity below 50%. To address these challenges, this study proposes a multi-objective multi-label feature selection model integrated with an optimized Fireworks Algorithm. The Improved Fireworks Algorithm Model incorporates Gaussian operators and adaptive functions while fusing fuzzy neural networks to enhance real-time threat response. Experimental validation across Palmer Penguin (small-scale), Fashion MNIST (medium-scale), and Bike Sharing (large-scale) datasets demonstrates three key advancements: Data processing capacity reaches 5,000 samples, exceeding Particle Swarm Optimization and standard Fireworks Algorithm baselines by 66%; Sensitivity maintains 70%-100% across datasets, outperforming traditional methods by 30% points; In a medium-sized data set, the research method scored only 5 out of 10 in the five indicators of comprehensive performance comparison based on the weighted geometric mean of the five-dimensional radar chart, indicating that the research method may have problems of overfitting or insufficient generalization ability when processing complex data. Adaptive adjustment time is reduced by 50%, confirming significant efficiency gains. These findings establish a robust framework for dynamic network security while highlighting scalability constraints in complex data environments.

**Keywords** Network security analysis, Feature selection, Improved fireworks algorithm, Multi-objective optimization, Fuzzy neural network

With the quick growth of the Internet and the advancement of the global informatization process, network security issues have gradually emerged, and network attacks can cause serious harm to personal privacy, business secrets, and national security, so protecting network security has become an urgent task. The methods of network attacks are constantly upgrading<sup>1,2</sup>. The attack methods used are becoming increasingly complex, including not only virus attacks but also emerging distributed denial of service (DoS) attacks. These attack methods pose significant challenges to network security and require technical and legal means to address them<sup>3</sup>. Furthermore, in the context of mobile Internet, individuals can access the Internet at any given moment and any location through the use of mobile devices, including smartphones and tablets. Moreover, rapid advances in cloud computing and big data have introduced new complexities into network security. Large-scale data leakage events will not only lead to personal privacy leakage, but also bring huge losses to enterprises and governments<sup>4</sup>. Therefore, it is crucial to strengthen the network security protection of mobile Internet<sup>5</sup>. However, traditional network security protection methods are inefficient, have poor defense performance, and low sensitivity. The real-time feedback mechanism is essential for monitoring and dynamically correcting model performance. By introducing feedback loops, the model parameters can be adjusted promptly based on the algorithm's performance in identifying malicious behavior, thereby improving its adaptability to new types of malicious behavior. The research goal is to build an improved fireworks algorithm model (IFWAM), aiming at the difficulties encountered by traditional fireworks algorithms (FWAs) in complex data processing and adaptive selection of high-quality explosion points. This improves the detection capability and response speed of the data security system, effectively protects against potential malicious behavior, and solves the limitations of FWA in complex data processing and adaptive selection of high-quality explosion points.

The main assumptions made in the research are that multi-objective optimization can effectively improve the accuracy of feature selection (FS), real-time feedback mechanisms can enhance adaptive capabilities, and combination algorithms can improve algorithm performance. Based on the above assumptions, the research has

State Grid Hubei Electric Power Research Institute, Hubei 430077, Wuhan, China. ✉email: hakt19810327@163.com

made the following contributions: (1) The algorithm combines an improved FWA and fuzzy neural network, providing a structural optimization method that improves data security and analysis efficiency through effective FS. (2) The concept of real-time feedback mechanism is proposed, and its applicability in dynamic network environments is explored. (3) The theoretical research on multi-objective optimization and fuzzy neural networks is enhanced, and new methods and technical support for data security solutions in practical applications are provided.

This study is divided into five sections. The first section introduces the research background and objectives. The second section summarizes domestic and international research achievements in network security analysis and feature selection. The third section presents the construction of the improved feature selection model combined with the fireworks algorithm. The fourth section provides the performance testing and application analysis of the improved algorithm. The fifth section concludes the paper and discusses future research directions.

## Related works

### Network security analysis methods

Network security is related to the confidentiality of a large amount of online data and the stability of online systems. Many scholars have conducted relevant research on network security analysis methods. Waqas et al. proposed a solution based on artificial intelligence and machine learning to address various security threats in wireless network security analysis methods. During the process, different types of security threats were identified and a classification system for artificial intelligence technology to address these threats was constructed. Results denoted that artificial intelligence could effectively enhance the security of wireless networks and address increasingly complex security threats<sup>6</sup>. Ping studied data encryption technology to address data protection issues in network information security analysis methods. During the process, the encryption and decryption processes of the algorithm were analyzed, and their advantages and disadvantages were compared. A hybrid model was formed by combining the algorithms. Comprehensive analysis showed that the proposed method could effectively ensure information security and has high confidentiality<sup>7</sup>. Zhao proposed a dot product algorithm that combines scalable video coding and sliding windows to address encryption technology issues in network security analysis methods. Critical analysis was conducted on other algorithms, pointing out the security risks associated with different fragile keys and proposing improvement plans. The experimental results showed that the algorithm improved efficiency and reduced computational and storage requirements<sup>8</sup>. Hong et al. proposed a new automated data auditing method to address the issue of inconsistent labels in network security analysis methods. Through experiments on real security operation centers and open-source datasets, it was verified that this data auditing method could identify erroneous labels and improve the accuracy of machine learning models through label correction<sup>9</sup>. Taheri et al. conducted a review of software defined network security issues, with a focus on analyzing the application of deep learning in software defined network security. The article first introduced the types of attacks faced by software defined networks and explored research on using deep learning algorithms to detect and mitigate these attacks. Research showed that deep learning methods could better identify complex attack patterns and achieve good detection results compared to other traditional network security analysis methods, such as statistical and threshold methods<sup>10</sup>.

### FS network related research

Some scholars have conducted relevant research on network FS. Thakkar and Lohiya reviewed intrusion detection systems using feature extraction techniques to address network security issues. They reviewed methods such as machine learning, deep learning, and swarm intelligence algorithms, emphasizing the importance of FS in models. Results denoted that feature extraction techniques were crucial for improving intrusion detection performance, providing guidance for research in the field of network security<sup>11</sup>. Rashid et al. proposed a tree-based stacked ensemble technique, combined with FS methods, to optimize the data scalability and complexity issues in network intrusion detection systems, improving the accuracy of the model. The experimental results showed that the proposed model outperformed existing methods in identifying normal and abnormal traffic, demonstrating its potential for application in the Internet of Things and large-scale networks<sup>12</sup>. Sah et al. proposed a model that combines FS methods and classifiers to address the FS problem of intrusion detection systems when dealing with large-scale network traffic. Research aimed to improve the detection performance of intrusion detection system by using FS techniques to remove irrelevant features and screen out features that have a significant impact on detection. The experimental results showed that the research method could significantly improve intrusion detection performance while reducing computational costs<sup>13</sup>. Hema et al. analyzed the diagnosis of Parkinson's disease using four feature extraction methods and classification algorithms. In the study, forward backward, rough set, and tree-based FS techniques were used, and compared with four classification methods: support vector machine, naive Bayes, K-nearest neighbor, and random forest. Research showed that the random forest algorithm, when combined with four FS methods, performed the best in accuracy<sup>14</sup>. Mounica and Lavanya proposed a high-performance computing model based on deep learning for traffic flow analysis of Twitter data. To address network security issues, they used feature extraction techniques to process Twitter data, including pre-processing of tweets and embedding vectors using unary, binary, and part of speech features. The experimental results showed that this method achieved the highest accuracy of 98.83% on the Kaggle dataset, outperforming other techniques<sup>15</sup>.

## Summary of the overview

The summary of relevant research and comparison with research methods are shown in Table 1.

In summary, a large number of studies have shown that FS technology can effectively improve the efficiency and accuracy of network security analysis by reducing the data dimension and eliminating redundant information, especially in intrusion detection, abnormal behavior recognition and other scenarios. However,

Related work's technique	Proposed method	Key differentiators with research method	Advantages of research method	Refs.
AI/ML for wireless threat classification	Multi-objective feature selection with real-time feedback	Dynamic adaptation (vs. static frameworks)	Higher sensitivity (70–100%) and real-time adaptability	6
Hybrid data encryption	Feature-based filtering + optimization	Optimizes preprocessing efficiency (vs. cryptographic focus)	Lower computational overhead & faster processing	7
SVC + sliding window dot-product	Non-encrypted dimensionality reduction	Reduces computational overhead (vs. encryption efficiency)	Better scalability to large datasets (up to 5000 samples)	8
Automated data auditing	Multi-label FS with feedback mechanism	Prevents label bias at feature level (vs. label correction)	Improved label consistency & reduced false positives	9
DL for SDN attack detection	Lightweight FNN + heuristic optimization	Lower complexity for real-time use (vs. deep learning)	Faster convergence & lower resource consumption	10
Feature extraction for IDS	Dynamic MOOMLFS with adaptive functions	Solves nonlinear interactions & local optima (vs. general FS)	Enhanced global search capability & avoidance of local optima	11
Tree-based ensemble + FS	Label-correlation-aware FS	Avoids ensemble computational cost (vs. stacked models)	Higher accuracy with reduced model complexity	12
FS + classifier combination	Anti-interference feature validation	Dynamic malicious feature filtering (vs. static selection)	Superior malicious behavior detection (83–95% accuracy)	13
Multiple FS methods for classifiers	Multi-objective label-structure optimization	Joint feature-label space optimization (vs. classifier-centric FS)	Better handling of multi-label correlations	14
DL feature extraction for Twitter	Gaussian-driven adaptive FS with fuzzy rules	Robust noise handling (vs. NLP engineering)	Improved noise robustness & interpretability via fuzzy logic	15

**Table 1.** Summary of related research and comparison with research methods.

existing studies have also revealed the key bottlenecks in the practical application of FS technology. First, in the multi-user collaborative environment, the traditional FS algorithm is difficult to balance the correlation between individual user characteristics and global data distribution due to the lack of dynamic adaptability, resulting in limited model generalization ability; Second, in high-dimensional data space, nonlinear interaction and noise interference between features exacerbate the computational complexity of the algorithm, and existing methods (such as filtering FS based on information entropy) are prone to local optimality, and it is difficult to take into account the robustness and interpretability of feature subsets<sup>16</sup>. On this basis, this study attempts to introduce optimization technology to optimize FS and design a better performance network security analysis method.

**Network security analysis of optimized feature selection algorithm and FWA**

This section is divided into two sub-sections to study network security based on FS and FWA optimization. The first sub-section is about the model construction of the proposed multi-objective optimization multi-label feature selection (MOOMLFS) algorithm and the data processing process of the proposed algorithm. The second sub-section is to study the core map and algorithm flow of the reconstructed neural network fusion FWA model.

**Multi-label feature selection algorithm model for multi-objective optimization**

When selecting features, the minimum feature set is found by reducing the dimensionality of the data, allowing the model to achieve optimal performance<sup>17,18</sup>. The single-objective FS algorithm faces a decrease in accuracy in dimensionality reduction of high-dimensional data space, which cannot meet the screening requirements for network security data. Therefore, an multi-label FS algorithm model for multi-objective optimization is constructed based on FS, as indicated in Fig. 1.

In the multi-label feature recognition model shown in Fig. 1, network data begins to flow after being preprocessed by the model. Some malicious behaviors and viruses also propagate with normal network data. The data is transmitted to the next stage for FS through the initialization support of the algorithm in the model. In this stage, in addition to the rough selection of the model, there is also advanced FS processing for some complex data information. After the feature recognition is completed, the interference factor system is used to match the recognized data features and perform anti-interference testing on the algorithm to check whether there are unsafe factors and malicious information in the selected data features<sup>19,20</sup>. The improved MOOMLFS model of the data feature archive is combined with data optimization to remove and process malicious information, and finally output optimized network data information. In a complex archive, the spatial distance between each data can be indicated by Eq. (1).

$$D_i = \sum_{j=1}^n (|f_j^{i+1} - f_j^{i-1}| + |f_2^{i+1} - f_2^{i-1}|) \tag{1}$$

In Eq. (1),  $D_i$  represents the distance between each piece of data, the spatial distance,  $i$  represents the member coefficient,  $j$  represents the objective function,  $n$  represents the archive coefficient,  $f$  represents the spatial framework, and the crowding distance between data information is represented by Eq. (2).

$$D_i = |f_1^{i+1} - f_1^{i-1}| + |f_2^{i+1} - f_2^{i-1}| \tag{2}$$

The classification method with a single-objective and label is not suitable for multi-label tasks, as there are correlations between interfering factors in multi-label datasets. Therefore, the individual definition of multiple labels can be expressed as Eq. (3).

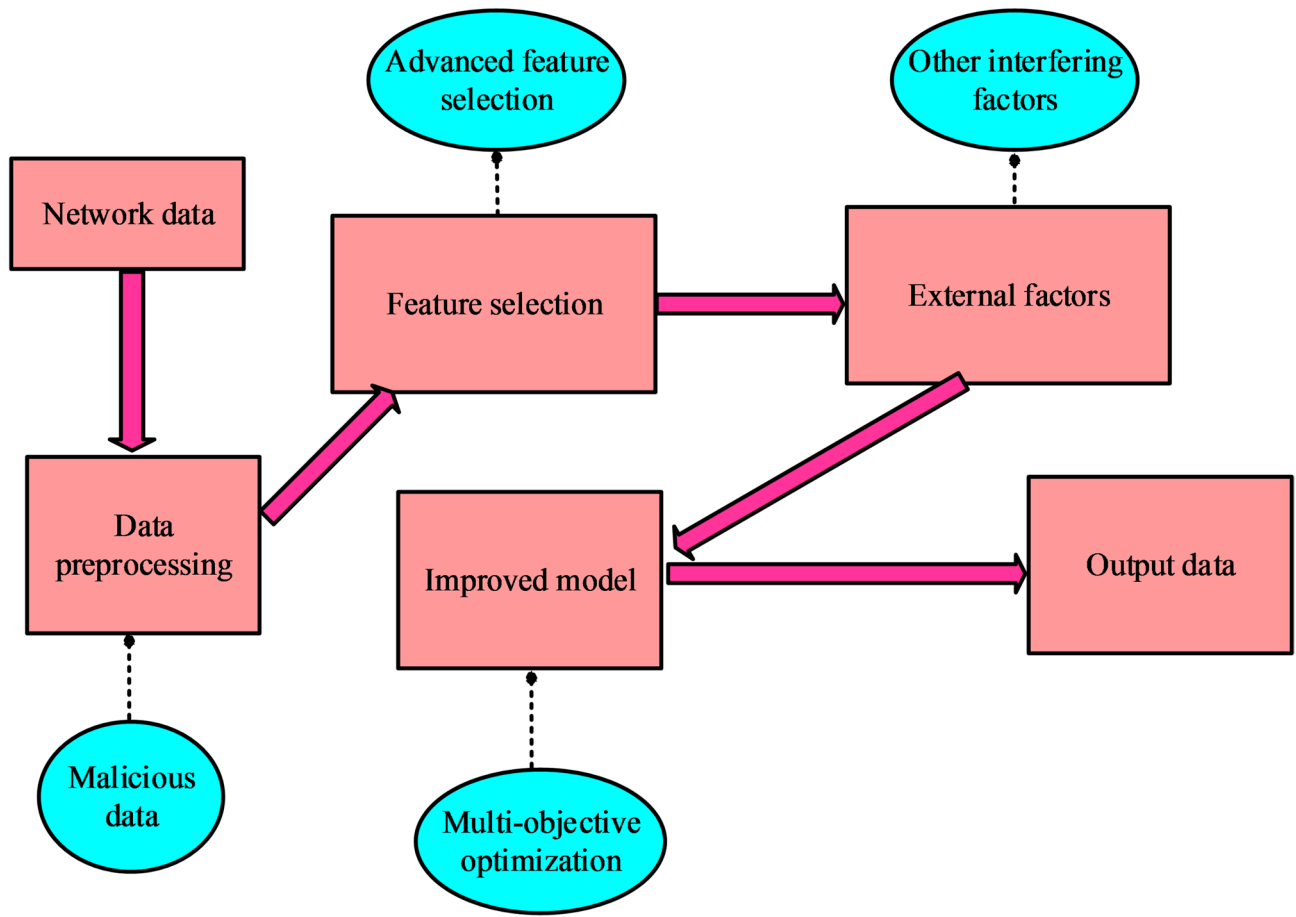


Fig. 1. Feature selection algorithm model for multi-objective optimization.

$$\begin{cases} X = \{x_1, x_2, \dots, x_m\} \\ Y = \{y_1, y_2, \dots, y_m\} \end{cases} \quad (3)$$

In Eq. (3),  $X$  represents the specific data information in the feature dataset,  $Y$  represents the data information in the interference dataset,  $m$  represents the maximum spatial data volume, and the label space can be represented by Eq. (4).

$$D = \{(x_i, y_i), 1 \leq i \leq m\} \quad (4)$$

In Eq. (4),  $y_i$  represents the actual label set of feature dataset  $x_i$ . The flowchart of the proposed algorithm model is denoted in Fig. 2.

In the MOOMLFS model of Fig. 2, there exists an original dataset. It assumes that there are  $N$  sample numbers in the dataset. The dataset can be divided into feature dataset and label dataset. It assumes that the number of feature datasets is represented by  $f$  and the number of label datasets is represented by  $y$ . After the algorithm starts running, the population is initialized first. After initialization is completed, the data is transmitted to a multi-label learner for algorithm training. The data information is divided into a training dataset and a testing dataset. The training dataset is trained using a multi-label learner, while the testing dataset is tested using a multi-label learner<sup>21</sup>. Then the results are calculated by the algorithm, updating the probability of data crossover and mutation. The archive database is updated through an MOOMLFS model designed for research. The algorithm evaluates whether the archive needs to meet the requirements. If it does not meet the requirements, it will return to the learning period for retraining and parameter optimization. If the requirements are met, it will output the archive and end the process. The predicted labels in the model can be expressed as Eq. (5).

$$RL(h, H) = \frac{1}{t} \sum_j \left\{ \frac{1}{y_i |y'_l|} \mid (k, l) \in (y_j \times y'_l), s.t. h(x_i, k) \leq h(x_j, l) \right\} \quad (5)$$

In Eq. (5),  $RL$  represents the metric function,  $h$  represents the training set learning function,  $H$  means the test dataset,  $t$  represents the training time,  $y_i$  represents the training dataset,  $x_i$  means the training data,  $x_j$  means

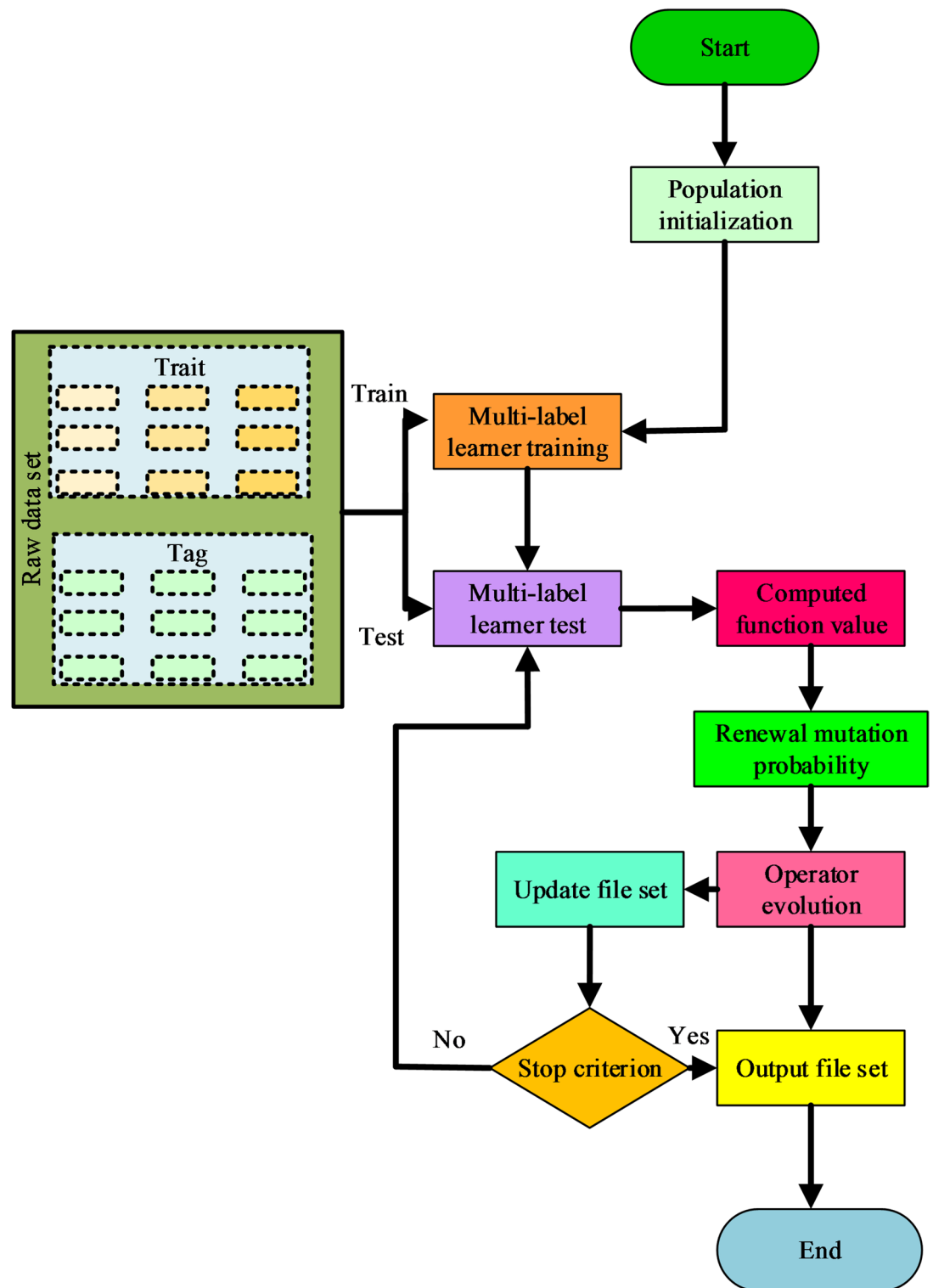


Fig. 2. Multi-label feature selection model for target optimization.

the true label,  $i, j$ , and  $l$  represent the data coefficients, and  $s.t$  means the combination of the training dataset and the test dataset. The calculation of tag ranking can be expressed as Eq. (6).

$$AP = (h, H) = \frac{1}{t} \sum_{j=1}^t \frac{1}{|y_i|} \sum_{y \in y_i} \frac{|\{y' | \text{rank}_n(x, y') \leq \text{rank}_n(x_i, y), y' \in y_j\}|}{\text{rank}_n(x_j, y)} \quad (6)$$

In Eq. (6),  $AP$  represents the label set of average samples above the specified label, and  $\text{rank}_n$  represents the ranking expression of the sample dataset. The minimum number of operations required to find the actual label through the process can be expressed as Eq. (7).

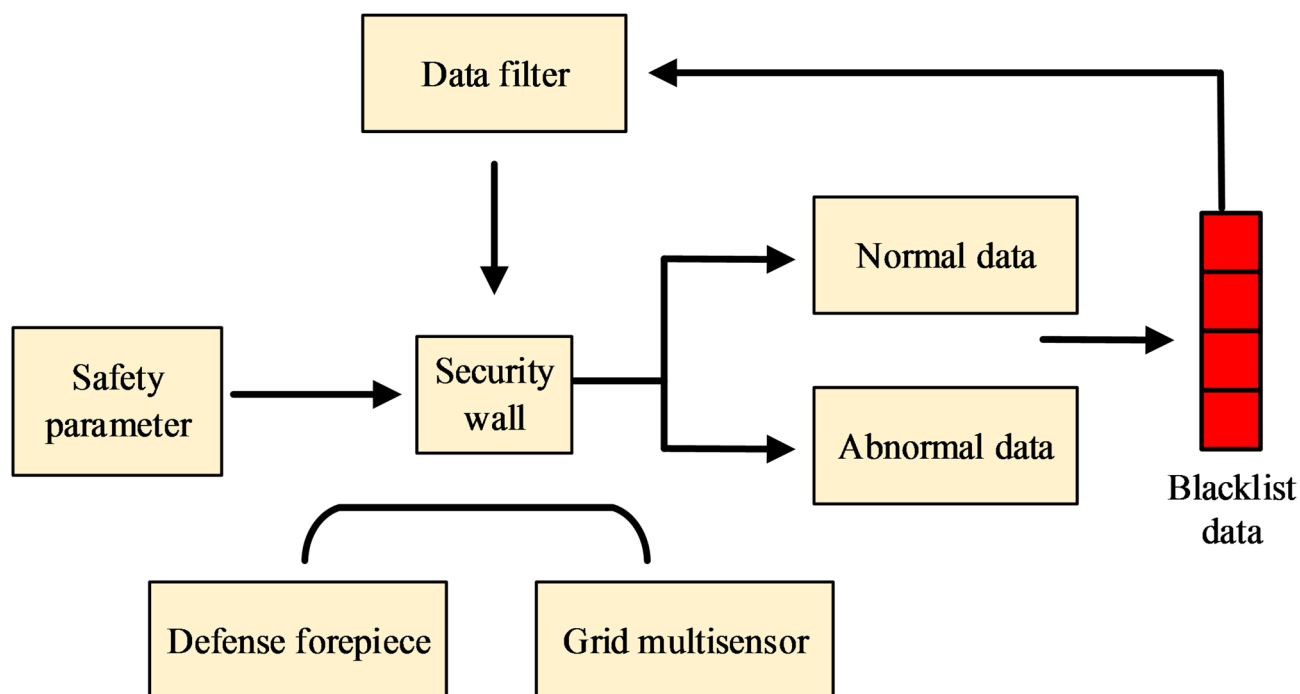
$$CV(h, H) = \frac{1}{t} \sum_{j=1}^t \max_{y \in y_i} \text{rank}_n(x_j, y) - 1 \quad (7)$$

In Eq. (7),  $CV(h, H)$  represents the minimum number of operations to find the actual label through the process. The number of error markers can be expressed as Eq. (8).

$$HL(h, H) = \frac{1}{t} \sum_{j=1}^t |h(x_i) \oplus y_i| \quad (8)$$

In formula (8),  $HL(h, H)$  represents the number of incorrect marks.  $\oplus$  represents the symmetric difference between the actual label and the classification label. The system results of model application and network data processing and security protection are shown in Fig. 3.

In the system protection structure of Fig. 3, massive data information in the network needs to be monitored by the security defense structure in the network processing system after primary processing and noise filtering. The security defense structure of the network includes functions that can identify malicious factors in data information and security parameter reference indicators. Through the screening of defense agencies, many malicious data information and hidden data are identified. This structure is called a defense wall structure with security parameter protection. Malicious data mainly refers to network threats including viruses and worms, Trojan programs, spyware, ransomware, and phishing data. These malicious data are usually propagated through network requests, and their behavioral characteristics are significantly different from normal data. For example, malicious data often send out a large number of requests in a short period of time or are accessed through abnormal ports. To identify malicious and normal data, the system continuously monitors the behavioral characteristics of data in the network, such as access frequency, request patterns, etc., and analyzes the anomalies of data behavior through specific algorithms. In addition, establishing a malicious data blacklist and a normal data whitelist can quickly screen for known malicious data by comparing the legality of the incoming data.

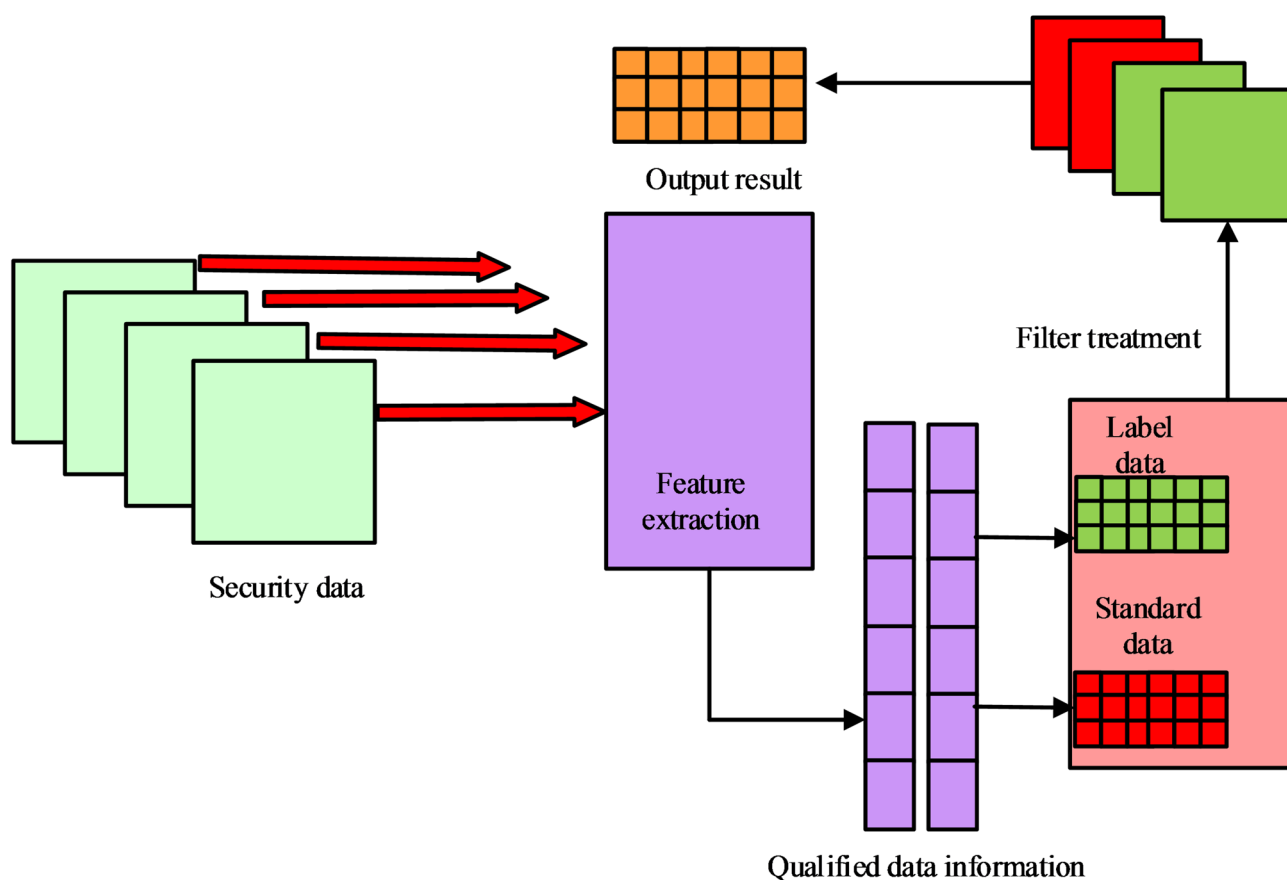


**Fig. 3.** Model application and network data processing and security protection system.

Ultimately, through multidimensional analysis and model optimization, it is possible to effectively distinguish between normal data and malicious data, ensuring network security and data integrity. The front half of the wall is a rigid wall protection mechanism that protects data information. The latter part is the structure of the sieve, which carries qualified labels through the grid structure of filtered data size based on safety monitoring data. If the data already has security labels but due to its large volume, it cannot flow through the grid structure to the next process. The multi-label FS model is the core of the grid structure, and only data that meets the feature requirements can be selected. Otherwise, it will be directly processed or returned for optimization<sup>22,23</sup>. Moreover, the information judged as abnormal data will be added to the blacklist, and the data in the blacklist will be filtered through a specific filter to identify the misclassified data information, ensuring the rigor of the program and algorithm. There is also an MOOMLFS model in the filter, and the enlarged network structure of the model in the filter and grid structure is shown in Fig. 4.

In Fig. 4, when the data information monitored through security passes through the grid structure in the back half of the protective wall, feature recognition follows the principle of dimensionality reduction for high-dimensional data, achieving data simplification and feature data information extraction. The extracted data feature information forms a feature data set, and due to the optimization of multi-objective recognition, the model can select features from data with a large amount of information in a shorter time. It compares and calculates feature datasets with multi-label datasets, leveraging the data connections between multi-label datasets to evaluate and calculate feature datasets. Individual data in the feature database that does not match the multi-label dataset is replaced with data from the label dataset to ensure that there is no significant difference between the data obtained through the model and the actual label dataset. The FS model in the filter is similar to that in the grid structure, because the filter needs to repeatedly filter, and the focus of the algorithm is on FS. The focus of the grid structure is not only on feature recognition and selection, but also on the standardization requirements of data information. The specific data characteristics are shown in Table 2.

From Table 2, there are seven feature types, including high-dimensional data dimension reduction features, feature dataset, multi-label dataset features, mismatched dataset features, filter model features, grid-structure model features, and multi-objective recognition features. The dimension reduction features of high-dimensional data are extracted by the dimension reduction technology. The main role is to simplify the data and extract the features. The feature dataset is a data set containing key information formed after dimension reduction, which is mainly used for subsequent feature identification and comparison. A multi-label dataset refers to a dataset containing multiple labels, and the main role is to evaluate and calculate feature datasets. Mismatched data replacement feature refers to the feature after the data that does not match the multi-label dataset is replaced. It

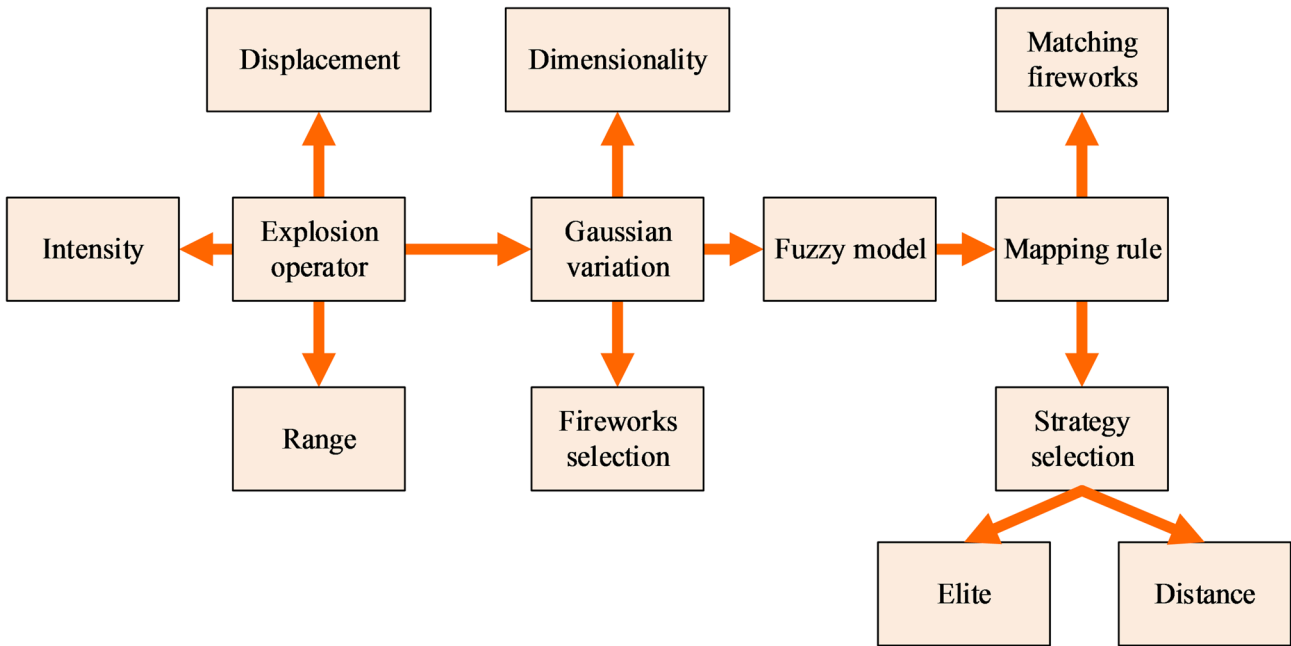


**Fig. 4.** The amplified network structure of the model in the filter and grid structure.



Feature type	Description	Function
Dimension reduction features of high-dimensional data	Data features extracted by dimension reduction techniques	Data simplification and feature extraction
Feature dataset	Datasets containing key information formed after dimension reduction	For the subsequent feature identification and comparison
Multi-label dataset characteristics	Dataset containing multiple labels	Assess and calculate feature datasets
Mismatched data replacement features	Features after replacing the data missing from the multi-label dataset	Ensure that the data is consistent with the actual labeled data set
Filter model features	FS model features used repeatedly in the filter	Filtering and feature identification
Grid-structure model features	The FS model features in the grid structure	Data standardization and feature identification
Multiple-objective recognition features	Optimized features for the rapid selection of key features from large amounts of data	Improve the identification efficiency and accuracy

**Table 2.** The feature types and descriptions of the feature extraction.



**Fig. 5.** IFWAM combined with fuzzy neural network model core mapping.

is mainly to ensure the consistency of the data with the actual labeled data set. Filter model features refer to the FS model features, repeatedly used in the filter, acting as filtering and feature recognition. Grid-structure model is the FS model feature in grid structure, which is data standardization and feature recognition. Multi-objective identification features means that the optimized features are used to quickly select the key features from a large amount of data, and the role is to improve the efficiency and accuracy of identification.

**IFWAM combined with fuzzy neural network model for core mapping**

By using an MOOMLFS model to train and test network data information, secure and standardized digital information is obtained. However, the traditional IFWAM faces difficulties in calculating fitness and optimal selection due to the complexity of data and the similarity of some explosion points. Therefore, the study optimizes the FWA and combines it with fuzzy neural networks to construct a fusion model for processing data information. The combination of fuzzy logic and IFWA is based on multiple core principles. Fuzzy neural networks quantify uncertainty through membership functions and convert continuous variables such as ‘strength’ and ‘displacement’ of explosion points into fuzzy sets, avoiding the limitations of binary decision-making. Fuzzy rule library dynamically guides the generation of fireworks explosion operators, replacing fixed threshold strategies. This mechanism enhances the algorithm’s global search capability in complex data spaces by adjusting the explosion radius and spark quantity in real-time<sup>24</sup>. The core orientation diagram of the fusion model is shown in Fig. 5.

In the fusion mechanism illustrated in Fig. 5, the FWA selects appropriate explosion operators based on the intensity, amplitude, and displacement of explosion points and generates multiple candidate explosion points through Gaussian mutation. To improve the accuracy of candidate selection, a fuzzy neural network is applied to support the decision-making process. The definition of fuzzy rules is based on three core input variables:



explosion intensity, displacement, and fitness value. Explosion intensity and displacement are divided into three levels, namely low, medium, and high, while fitness is divided into poor, good, and excellent. A fuzzy rule base is then constructed according to these levels. When the explosion intensity is high, the fitness is excellent, and the displacement is small, the selection probability of the candidate point is very high. When the three variables are at a medium level, the selection probability is medium. When the intensity is low, the fitness is poor, or the displacement is large, the selection probability is low. These rules are processed through a Mamdani-type fuzzy inference mechanism to map input fuzzy sets to output selection probabilities, providing an initial decision basis for candidate points.

The quantification standard of the proximity principle is subsequently introduced. It evaluates the geometric distance between each candidate point and the global optimal point and converts the distance value into a weight coefficient between 0 and 1 to represent the degree of proximity. A shorter distance corresponds to a larger weight, while a longer distance corresponds to a smaller weight. This weight is then multiplied by the selection probability obtained from fuzzy inference, and all selection probabilities are normalized to ensure that their sum equals one. Through this process, the heuristic screening of fuzzy rules and the quantitative correction of the proximity principle are effectively integrated, which enhances global exploration capability, improves local exploitation accuracy, prevents premature convergence, and increases robustness and adaptability in complex network environments. To obtain better fireworks positions, the initial fireworks are discretized to better cover and seek the optimal solution. The coverage length can be expressed as Eq. (9).

$$l = x_{\max} - x_{\min} \quad (9)$$

In Eq. (9),  $x_{\max}$  and  $x_{\min}$  denote the max and the min values of the coordinate space, respectively. The coverage length in a certain dimension can be expressed as Eq. (10).

$$l' = (x_i)_{\max} - (x_i)_{\min}, i \in [1, k] \quad (10)$$

In Eq. (10),  $(x_i)$  represents the  $i$ th dimensional coordinate in space, while  $(x_i)_{\max}$  and  $(x_i)_{\min}$  respectively represent the max and mini values on the  $i$ th dimensional coordinate. Fuzzy neural network is a theory of fuzzy mathematics, where one data corresponds to one set, and there are only two ways: belonging and not belonging. Fuzzy sets can be expressed as Eq. (11).

$$A(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A \end{cases} \quad (11)$$

In Eq. (11),  $A$  represents the set domain,  $x$  represents the elements, and the extension of fuzzy relationships can be expressed as Eq. (12).

$$U \times V = \{(x, y) | x \in U, y \in V\} \quad (12)$$

In Eq. (12),  $U$  represents two ordinary sets, and when  $U = V$  occurs, the fuzzy set is referred to as the fuzzy relationship of ordinary sets. The flowchart of the model is shown in Fig. 6.

In the model processing flow of Fig. 6, the data is initialized first, and an initial fuzzy model is established using the initial explosion point of the IFWAM. Then, the initial fireworks population is generated based on the encoding, which is called data initialization. It should determine whether the initialized data meets the termination conditions. If conditions are met, the process should be brought to a conclusion. Conversely, if the conditions are not met, the process should be continued in a downward direction. Fireworks continue to be de-coded as a precursor to the fusion FWAM, and the algorithm is fused with the fuzzy neural model based on the similarity of fuzzy rules and fuzzy sets to prevent rejection reactions<sup>25,26</sup>. Next is to identify the parameters of the components and calculate the fitness function. Sparks are generated based on the IFWAM, individual fireworks are selected, and the population of fireworks is optimized. The FWA can calculate the fitness value of the population, and only when there is a reasonable fitness value can the algorithm generate explosive sparks. The spark of algorithm explosion can be expressed as Eq. (13).

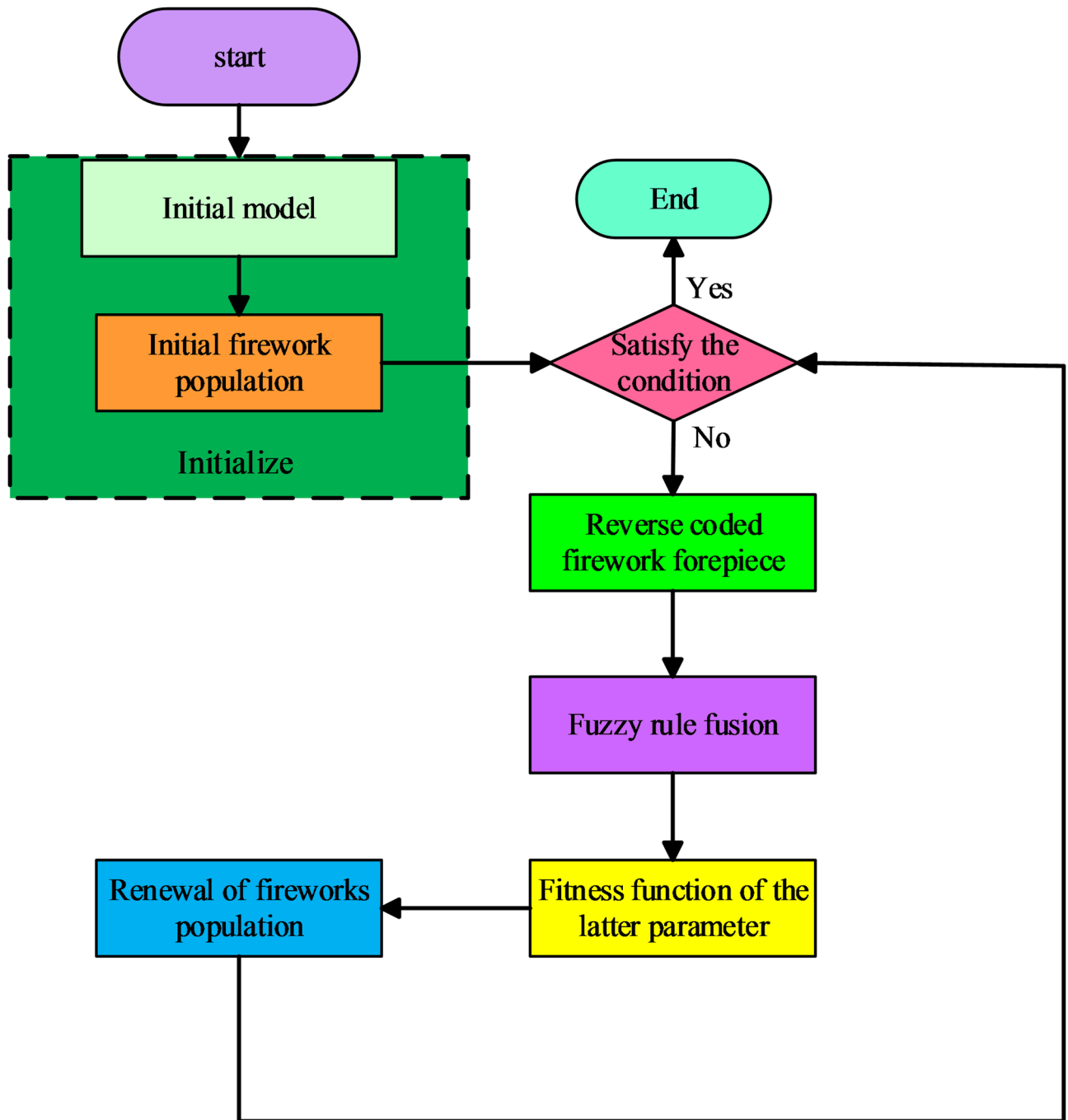
$$H_i = H \cdot \frac{L_{\max} - L(x_i) + I}{\sum_{i=1}^N (L_{\max} - L(x_i)) + I} \quad (13)$$

In Eq. (13),  $H_i$  represents the degree of algorithm explosion,  $H$  represents the explosion control parameter,  $L(x_i)$  represents the fitness value of the algorithm,  $L_{\max}$  represents the maximum fitness value of the algorithm, and  $I$  represents a constant. The limitation of explosion sparks can be expressed as Eq. (14).

$$S_i = \begin{cases} \text{round}(\alpha \cdot s) \\ \text{round}(\beta \cdot s) \end{cases} \quad (14)$$

In Eq. (14),  $\alpha$  and  $\beta$  represent the set spatial parameters, and  $\alpha < \beta$  and  $\text{round}()$  are integer functions. The satisfaction conditions of other parameters are represented by Eq. (15).

$$\begin{aligned} S_i &< \alpha, \beta \\ S_i &> \alpha, \beta \end{aligned} \quad (15)$$



**Fig. 6.** The flow chart of IFWAM combined with fuzzy neural network model.

The FWA is improved by calculating the fitness value and modifying the parameters of the generated initial fireworks population through a fitness function. The added Gaussian operator improves the randomness of the fitness function for solving the optimal fireworks, and the explosion rule of the improved algorithm is shown in Fig. 7.

In the explosion rule shown in Fig. 7, there are certain rule restrictions during the processing of the input data by the FWA, and the chaotic and massive network data is processed through the spatial dimension of the FWA. Based on the initial explosion point selected by the algorithm, the data is parameterized and analyzed. By calculating the fitness function and Gaussian operator, the data of adjacent explosion points can be randomly remembered and recognized for their features. It matches data points with high similarity to the initial explosion point and without carrying malicious information for memory and storage. Due to the complexity of network data, data information can be divided into two parts during initial processing, and processed through a grid structure similar to FS. The core of the actual grid structure is an IFWAM. After processing, the two network data still remain in the spatial dimension of the FWA, and the selected suitable data points similar to the initial

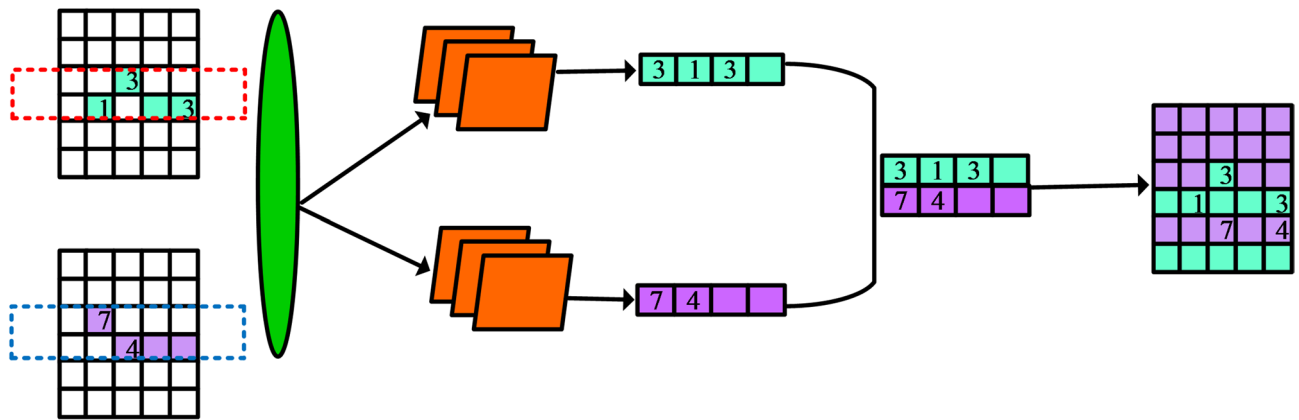


Fig. 7. The explosion rules of IFWA.

#### Algorithm 1 MOOMLFS

**Input:** Dataset D containing features and labels

Target Variable T

Number of selected features N

**Output:** Subset of features  $F_{\text{selected}}$

**Initialize:**  $F_{\text{selected}} = []$  # List to hold selected features

$F = \text{ExtractFeatures}(D)$  # Extract features from dataset D

$L = \text{ExtractLabels}(D)$  # Extract labels from dataset D

**For each feature f in F:** a. Compute relevance  $R(f, T)$  using a chosen metric (e.g., correlation, mutual information)

Sort features F by relevance R in descending order

**Select top N features:**  $F_{\text{selected}} = \text{TopNFeatures}(F, R, N)$

**Return**  $F_{\text{selected}}$

Table 3. The pseudocode for the proposed method.

fireworks explosion point are combined together to form a new dataset<sup>27</sup>. The dataset optimizes and adjusts itself by combining and comparing features with each other, and ultimately outputs results in the form of combined data information. The pseudocode for the proposed method is shown in Table 3.

#### Performance and application analysis of improved feature selection algorithm and FWA

This section is divided into three sub-sections to test the efficacy of the algorithm. The first sub-section is a parameter setting table for testing the performance of the IFWA. The second sub-section is a performance test of the improved feature selection algorithm (FSA) and FWA. The third sub-section is about the analysis of the effectiveness of improving the algorithm in practical applications.

##### Improved feature selection algorithm and IFWAM performance testing parameter settings

Optimizing the FSA into multi-objective feature recognition can improve the processing speed of the algorithm for network data. Meanwhile, the IFWAM can solve for the optimal explosion point. The related parameter settings during the optimization of the FWA are denoted in Table 4.

From Table 4, in the optimization, the optimal size of the fireworks population was 50, and the optimal number of iterations and tests for the algorithm was 150, which could obtain the optimal weight and bias values of the algorithm<sup>28</sup>. After combining the improved algorithm with the fuzzy neural network model, to avoid model collapse caused by incompatibility between the two algorithms, a fusion threshold similar to fuzzy sets and rules was set to protect the hardware devices implementing the algorithm by limiting the threshold range. The optimal fusion threshold was 0.5 and 0.7. There were weighting factors in the fitness function of the improved algorithm. The weighting factors for the fitness function could be set to 0.7, 0.1, and 0.1 after multiple calculations. There was also a certain threshold for the population size of the FWA, with the best dataset being 85 and the best fireworks population being 50. The principles, categories, and impacts of malicious attack behavior during FS are denoted in Table 5.

Parameter name	Value	Parameter number
Number of optimization Iterations	150	1
Fitness factor	$X_1=0.7, X_2=0.2, X_3=0.2$	2
Fuzzy rule similarity convergence threshold	0.5, 0.7	3
Parameter setting	$N=85, A=50$	4
Firework population size	$N=50$	5

**Table 4.** Algorithm performance test parameter setting table.

Attack category	Impact	Principles
Local attacks	Grants unauthorized intruders operational privileges	Manipulates the host remotely through malicious code
Remote attacks	Illegal occupation of system resources	Elevates account privileges or steals account information through illicit means
DoS	Host is unable to perform operations or crashes	Floods the host with data, reaching storage or transmission limits
Eavesdropping attacks	Monitoring legitimate user operations	Attacks exploiting port vulnerabilities
Secret Attacks	Leakage of user privacy	Intercepts information and decodes keys
Man-in-the-middle attack (MITM)	Alters information transmitted between user and service	Intercepts and modifies communication between the user and the target server
Malware propagation	Spread of viruses or Trojan programs	Guides users to download malicious programs through emails or download links
Phishing attacks	Deceives users into entering sensitive information	Creates fake web pages to lure users into providing passwords and personal information

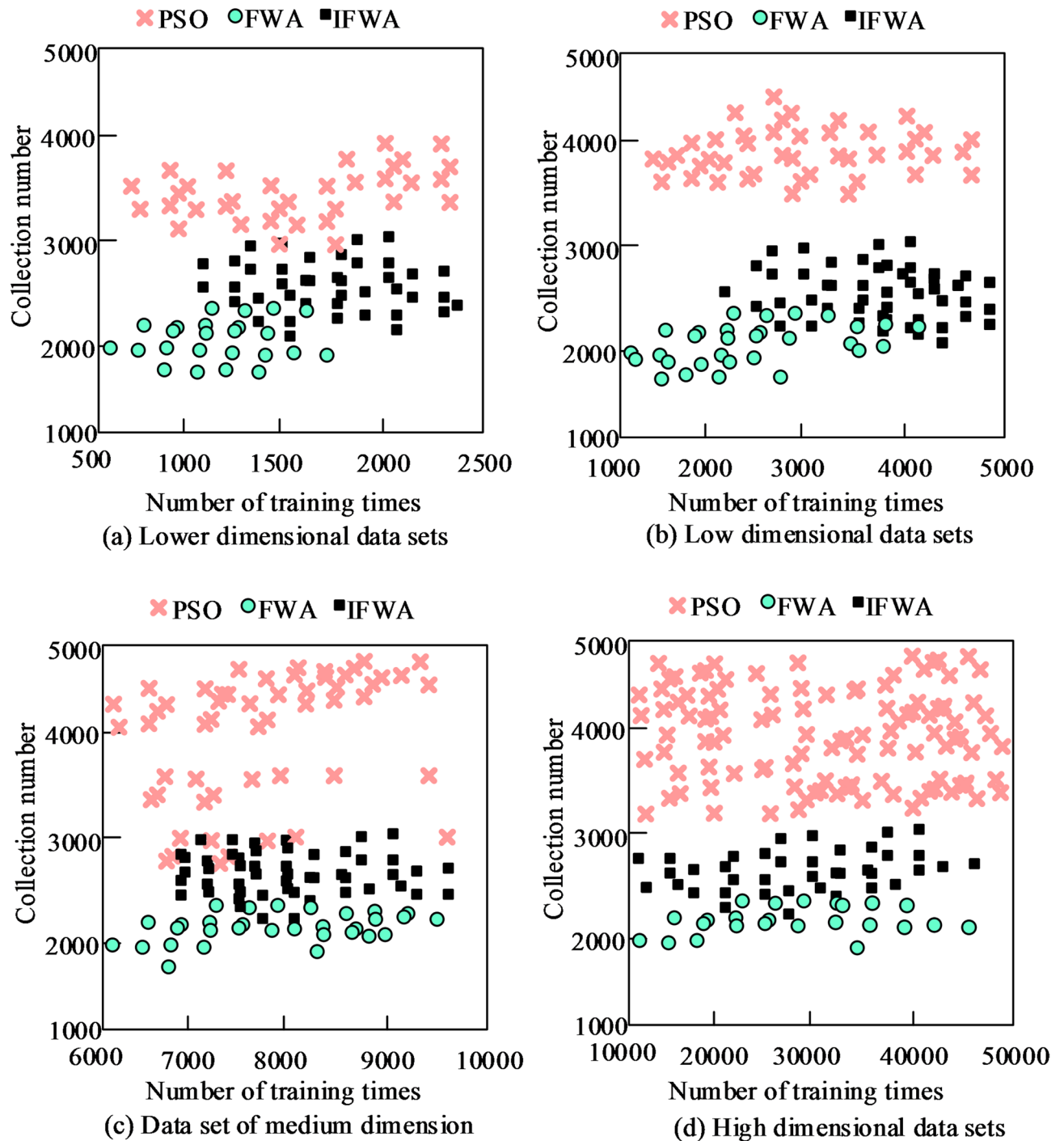
**Table 5.** Table of malicious network attacks.

In Table 5, during the optimization of the FSA, the occurrence of malicious behaviour, such as a DoS attack, could have a detrimental impact on the host’s storage capacity and transmission efficiency. This was achieved by the malicious actor sending substantial amounts of data, thereby overwhelming the host’s transmission capabilities and leading to a decline in its operational capacity. Consequently, the host might become unresponsive, ultimately resulting in a complete system failure. Local attack behavior utilized malicious code to randomly arrange and combine many letters and numbers to obtain the host’s password. After opening the system with the password, remote operations could be performed on the host. Such malicious behavior is a tacit approval of granting unauthorized intruders operations, resulting in the lack of security for user and host information. Remote attacks use illegal means to steal advanced users’ account information or change users’ account information, which can pose a threat to user privacy. Listening is the act of stealing user information through the use of network vulnerabilities, resulting in the leakage of users’ intentions and a decrease in trust in the algorithm. In standardized indicators, Accuracy represents the proportion of correctly identified malicious behavior samples to the total sample; Sensitivity/Recall represents the proportion of correctly identified malicious behavior, reflecting the risk of missed detections; Response Time represents the average time taken from data input to model output of safety analysis results, including the entire process of feature selection and classification inference<sup>29</sup>.

**Performance and effect analysis of the improved algorithms**

The parameter settings in Table 4 during the optimization of the FWA can ensure that the most reasonable training coefficients can be used to achieve the optimal weights and biases of the fireworks fusion model. Table 5 summarizes the common but ineffective malicious behaviors encountered by FSAs when processing network data. The study selected different datasets to test the performance of the improved algorithm. The smaller dataset used in the test was the Palmer Penguin dataset, the medium-sized dataset was the Fashion MNIST dataset, and the large dataset was the shared bike dataset. The link to the Palmer Penguin dataset was [https://gitcode.com/gh\\_mirrors/pa/palmerpenguins](https://gitcode.com/gh_mirrors/pa/palmerpenguins). The link to the Fashion MNIST dataset was [https://gitcode.com/gh\\_mirrors/fa/fashion-mnist](https://gitcode.com/gh_mirrors/fa/fashion-mnist) Fashion MNIST. The link to the shared bike data set was <https://github.com/topics/bikesharing>. The improved FWA collects feature information of data in different dimensions of the dataset, as shown in Fig. 8.

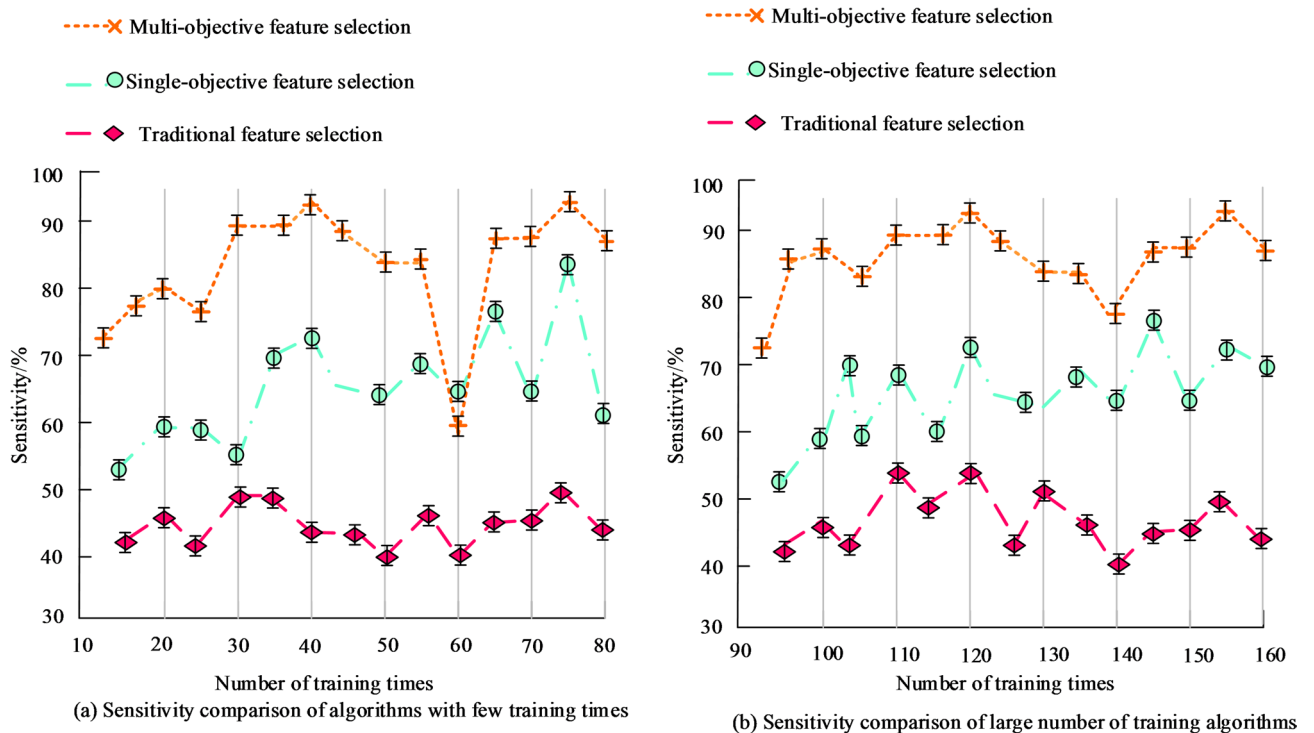
From Fig. 8, when the size of the dataset was small, the particle swarm optimization (PSO) algorithm collected data features more frequently when the training times were between 500 and 1700, but the number of collected data was relatively small, all below 2500. As the size of the dataset increased, the PSO algorithm still maintained a relatively low level of information collection for the dataset, but was no longer limited to a few training iterations for data collection. When the FWA was trained on smaller datasets with fewer iterations, the sensitivity of the data was lower and there was almost no processing of the dataset. As the size of the dataset increased, there was no significant change in the number of information collected by the algorithm, which remained below 3000. However, there was a significant change in the algorithm when the training times were greater than 10,000 and it was in a large-scale dataset. When IFWAM collected information from datasets, regardless of the size of the dataset or the amount of algorithm training iterations, the number of collected data information was almost always greater than 3000. Moreover, with the increase of data size and algorithm training times, it was evident that the IFWAM algorithm enhanced its ability to collect data information, with a wider range and more



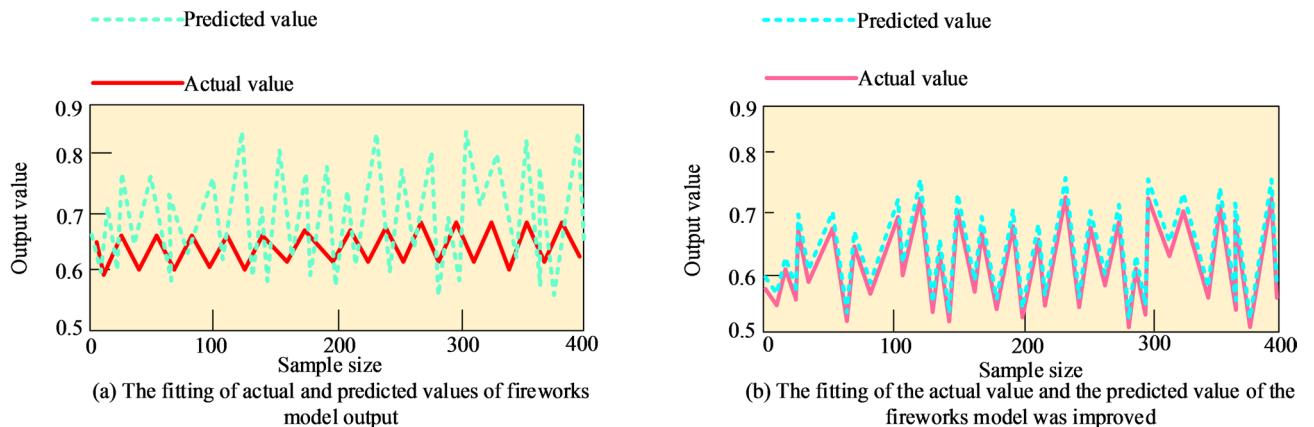
**Fig. 8.** The number of data that a feature selection algorithm can accurately select in data sets of different sizes.

numbers of data information. When the dataset size was intermediate or large, IFWAM's data collection reached 5000. The sensitivity of the feature algorithm varied with the number of training iterations, as shown in Fig. 9.

From Fig. 9, in Fig. 9 (a), when the training frequency was 20 times, the sensitivity of the traditional FSA was 45%, and the sensitivity of the single-objective FSA was 60%. The sensitivity of multi-objective algorithm was 80%. When the training frequency is 30, the sensitivity of the single objective algorithm is still between 50% and 60%; The sensitivity of multi-objective algorithm is still the highest, reaching 90%, and remains around 90% when the training frequency is in the range of 30 to 80. When the training frequency was 40 or 50 times, there was no significant difference in algorithm sensitivity with increasing training frequency. When the training frequency was 60, the sensitivity of the traditional FSA was 40%, and the sensitivity of the single-objective FSA was 64%. At this point, the sensitivity of the multi-objective FSA decreased to 57%. As the training frequency of the algorithm changed, the sensitivity of the traditional FSA fluctuated between 40% and 50%, the sensitivity of



**Fig. 9.** The sensitivity of different data feature recognition algorithms changes with the number of training.



**Fig. 10.** Improvement of the fit between the output prediction and the actual value of the FWA fusion model.

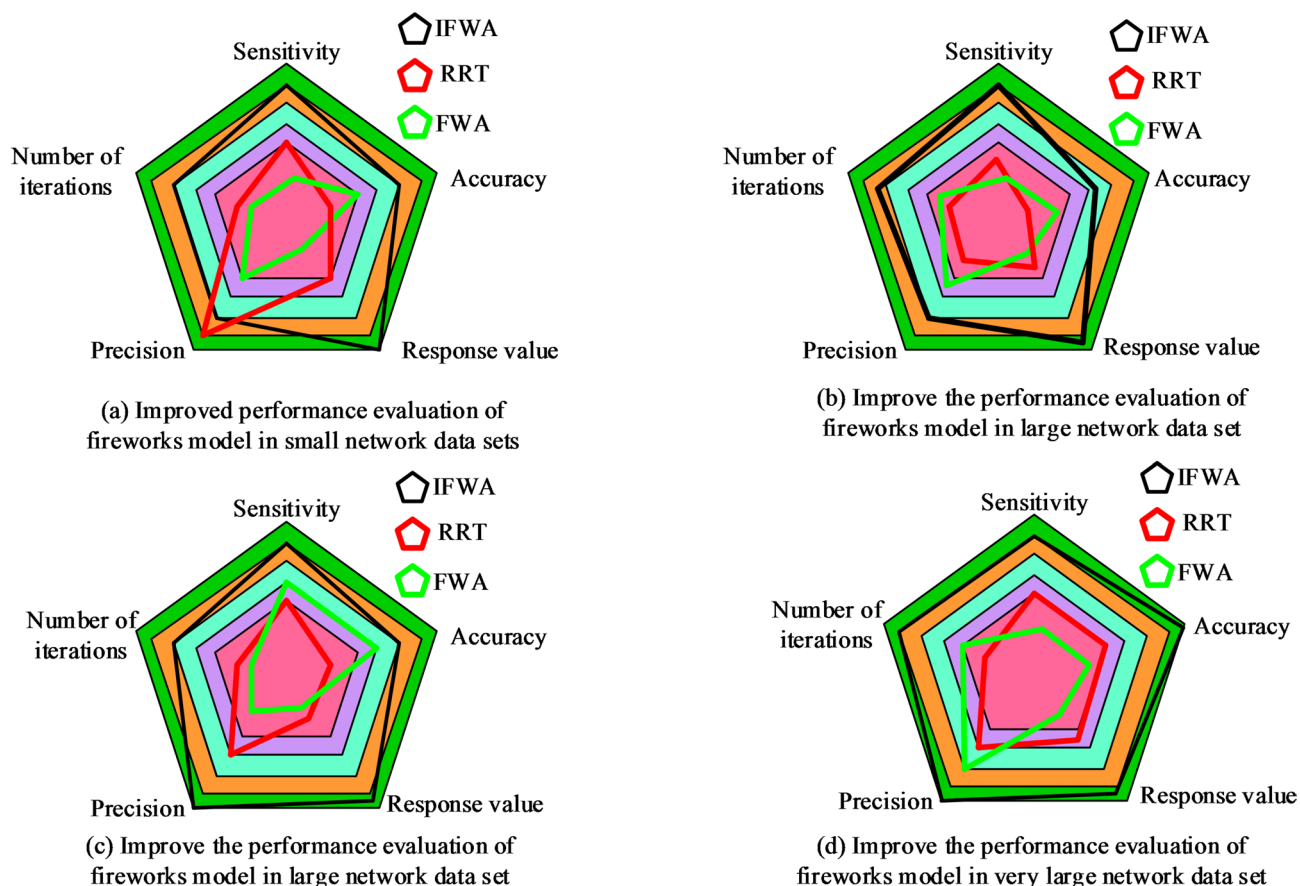
the single objective algorithm fluctuated between 50% and 70%, and the sensitivity of the multi-objective FSA fluctuated between 70% and 100%. Although the sensitivity of multi-target FSA reaches 70%–100%, there is an abnormal decrease at 60 training iterations. Through tracing, this phenomenon originated from the local feature similarity between fashion accessories and malicious traffic in the Fashion MNIST dataset, leading to false activation of the fuzzy neural network. This reveals the allergy problem of the model to specific non-threatening patterns. The fitting between the output prediction and actual values of the IFWAM fusion model is shown in Fig. 10.

From Fig. 10, the difference between the predicted output value and the actual value of the FWA model was significant, indicating poor fitting. The predicted values of IFWAM almost overlapped with the actual output line, indicating a high degree of fit. In 10 (a), when the sample size was 100, the actual and predicted output values of the FWA were 0.6 and 0.8, respectively. When the sample size became 200, the predicted and actual output values of the algorithm were 0.55 and 0.6, respectively. As the sample size increased, the predicted and actual output values of the algorithm respectively fluctuated between 0.53–0.83 and between 0.6–0.7, with a fitting degree of about 46.8%–54.5%. In 10 (b), when the training frequency was 100, the predicted and actual values of IFWAM were 0.6 and 0.5 respectively, with a difference of only 0.2 between both values. When the training frequency was 200, the predicted value almost matched the actual value. As the training frequency increased, there was



no significant difference between both values of the algorithm, with a maximum difference of 0.3. When the training times was 300, the actual and predicted values were 0.55 and 0.56, respectively. When the training times was 400, the predicted and actual values were 0.5 and 0.5, respectively. The long-term accumulation performance of IFWAM in processing network data information in different datasets in experimental environments is shown in Fig. 11.

The comprehensive performance score is based on the weighted geometric mean of five dimensional radar images, covering accuracy, sensitivity, response time, iteration times, and resource consumption, with a maximum score of 10 points for each item<sup>30</sup>. Figure 11 shows the comprehensive performance of different models in processing data information in long-term experimental environments on datasets of different scales. Study used the IFWAM combined with fuzzy neural networks to achieve accuracy, precision, and other metrics. A pentagon in Fig. 11 represents the scoring criteria of 2 points, and performance evaluation is represented in the form of pentagonal indicators. In a smaller dataset, the sensitivity score of the IFWAM was 8, indicating its excellent performance in identifying positive instances and effectively reducing false negative cases. However, its accuracy was 6, although it performed well, its accuracy dropped to 5 in medium-sized datasets, indicating that the model may have overfitting or insufficient generalization ability when dealing with complex data. In addition, a response time score of 10 demonstrated the efficiency of IFWAM in real-time applications, ensuring the feasibility of timely decision-making. If the number of iterations was 6, it indicated the robustness of the model in terms of convergence and the ability to reach a solution quickly. The performance of the rapidly-exploring random trees (RRT) model and the FWA model was worse than that of IFWAM. From the performance comparison chart, the indicators of the other two models were almost surrounded by IFWAM. Only the RRT model performed better than IFWAM in terms of accuracy performance, with a score of 8, but its accuracy and response values were poor, only 1, so the overall performance of the model was poor. The accuracy and precision of the FWA model were relatively good compared to its other performance, at 2, while the other performance evaluations were only 1. The testing accuracy of IFWAM significantly decreased to 5 on medium-sized datasets. This phenomenon indicates that the model may have overfitting or insufficient generalization ability when facing more complex data. In depth analysis shows that the performance degradation is mainly due to two reasons: the imbalance between model complexity and data size, and the insufficient sample size provided by medium-sized datasets to support the full learning and generalization of the IFWAM model's complex parameter space, which may result in overfitting on training data and poor performance on unseen data. Similar analysis results also appeared in the study of Chatur N et al., who found that in the resource allocation scenario of data transmission,



**Fig. 11.** Comparison of the integrated performance of the fusion model of FWA in different data sets with other models.



the accuracy of FWA decreased by more than 10% when the sample size was insufficient<sup>31</sup>. The complexity of nonlinear feature interactions increases, and medium-sized data typically contains richer nonlinear interaction relationships between features. The current structure or optimization strategies of IFWAM models have limitations in effectively capturing and processing such complex nonlinear patterns.

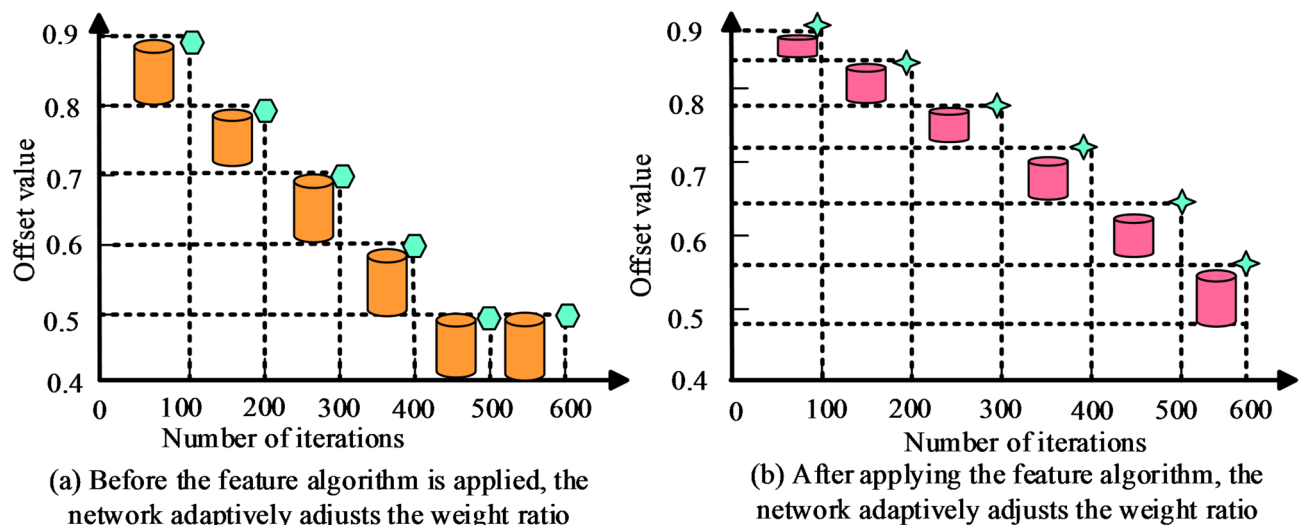
In terms of computational complexity, compared with traditional single-objective feature selection algorithms, the research method has increased the training time by approximately 15% to 20% and the memory usage by about 25%. This is mainly due to the overhead of multi-label evaluation and iterative calculation of adaptive functions. However, tests on large datasets such as the shared bike dataset show that while the model maintains an accuracy rate of 83% to 95% in identifying malicious behaviors, the growth of its resource consumption remains within a controllable range. Compared with the PSO and RRT algorithms, this model takes 1.3 times and 1.5 times respectively to complete the processing of 5,000 pieces of data in the same hardware environment, but the recognition accuracy has increased by more than 30%. Considering the parallel computing capabilities of modern server clusters, this level of resource growth is acceptable for real-time network security monitoring systems. In the future, resource efficiency can be further optimized through algorithm simplification and hardware acceleration.

### Analysis of the application effect of the improved algorithms

Performance experiments were conducted on the skill algorithm and model in different datasets, and the results demonstrated the stability of the improved algorithm's overall performance, as well as its high sensitivity to different types of data. To further confirm the feasibility of the improved algorithm, the study conducted experiments on the high-performance improved algorithm under simulation conditions. The time required for adaptive adjustment of network data before and after applying the FSA is shown in Fig. 12.

From Fig. 12, as the complexity of network data changed, the parameter bias value of the algorithm continuously decreased. As the number of iterations and the amount of network data changed, the optimal bias value of the algorithm was 0.4. In Fig. 12 (a), when the FSA was not applied, the adjustment time required from high bias values to low bias values was a training time of 0.1 bias values. As the complexity of the network data increased, the adaptive algorithm adjustment time did not decrease and still maintained the training time of 0.1 bias values until the number of iterations reached 500, which was already the optimal bias value of the algorithm. So the subsequent adjustment of bias values was no longer meaningful. In Fig. 12 (b), after applying the FSA, the adjustment time required from high bias values to low bias values was 0.05. As the complexity of the data increased, the adjustment time did not change. The adaptive adjustment time was reduced by half, and the data processing efficiency was significantly improved after the algorithm was applied. The comparison of perceived quality of different models facing network data information is shown in Fig. 13.

From Fig. 13 (a) and (b) show the variation of perceived quality size with data information size when applying a single-objective model and a multi-objective model, respectively. Algorithmic perception is essentially a feedback mechanism used to correct and monitor the performance of a model. When the size of the data information was 5, the perceptual quality of the single-objective model was 0.5, and the perceptual quality of the multi-objective model was 0.65, with a difference of 0.15. When the size of the data was 10, the perceptual quality of the single-objective model for data information decreased to around 0.38, and the perceptual quality of the multi-objective model also changed, with a change in its ability to perceive data information, decreasing to 0.54, with a difference of 0.16. When the data size was 15, the perceptual quality of the single-objective model improved to around 0.57, while the perceptual ability of the multi-objective model was 0.02 lower than that of the single-objective, at 0.55. When the data information size was 20, the perceptual quality of the single object



**Fig. 12.** Time comparison of adaptive adjustment of network data analysis before and after application of feature selection algorithm.

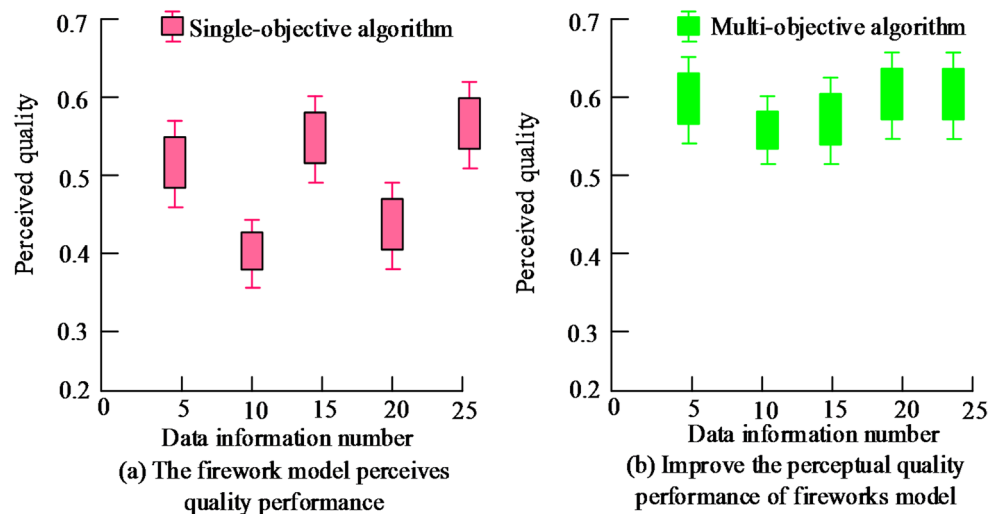


Fig. 13. Comparison of perceived quality of different models in network data information perception.

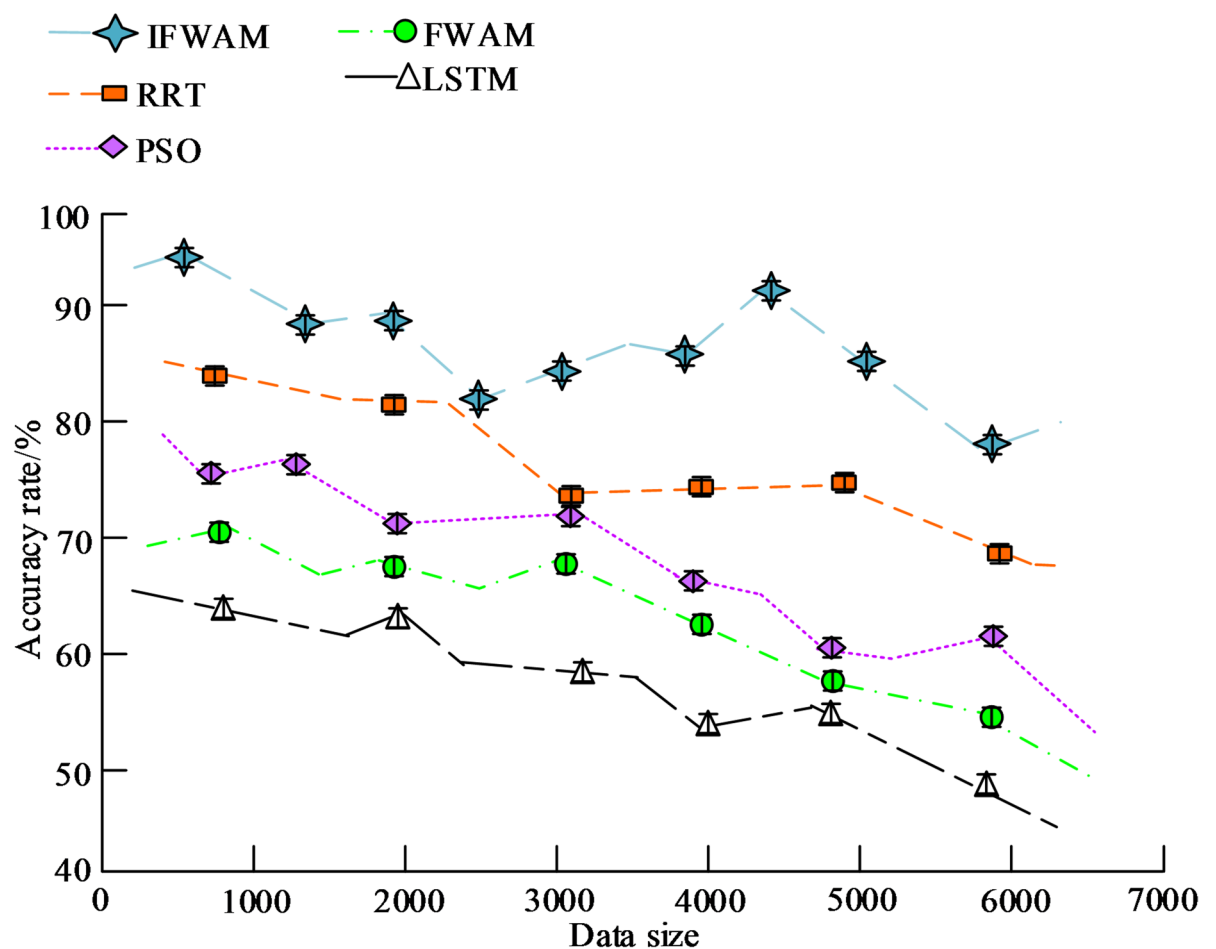


Fig. 14. The comparison of the identification accuracy of different algorithms for malicious behaviors in the same data set.

model was 0.4, and it was 0.6 at 25. The difference in perceived quality between multi-objective and single-objective was 0.1 and 0.03, respectively. The vast dataset not only contained normal data information, but also some malicious behaviors hidden in network data. The comparison of the recognition accuracy of different algorithms for malicious behaviors in the same dataset is shown in Fig. 14.

Metric	IFWAM	Standard FWA	PSO	RRT	LSTM
Average Threat Detection Rate (%)	95.8	82.3	85.6	79.1	93.5
Known Attack Detection Rate (%)	99.2	95.5	96.8	90.2	98.5
Unknown Attack Detection Rate (%)	73.5	45.1	48.3	41.6	65.2
Average False Positive Rate (%)	1.5	4.8	3.9	7.2	2.8
System Response Delay (ms)	135	208	192	185	350
CPU Utilization (Peak, %)	55	70	65	60	85
Feature Dimension (After Optimization)	45	72	68	–	–

**Table 6.** Comprehensive performance comparison of different algorithms in a simulated enterprise network security monitoring scenario.

Figure 14 compares the accuracy of malicious behavior recognition in network data information using five algorithms: PSO, RRT, Long Short Term Memory (LSTM), FWAM, and IFWAM. From the figure, the accuracy of different algorithms in identifying malicious behavior gradually decreased as the data sample size increased. When the sample size was 1000, IFWAM had the highest accuracy in identifying malicious behavior, with an accuracy of 95%. The accuracy of FWAM was 87%, while the RRT algorithm had a lower accuracy of around 74%, the LSTM algorithm had a slightly lower accuracy of 68%, and the PSO had the lowest accuracy of around 63%. As the sample size increased, the accuracy of recognition decreased, with IFWAM dropping from 95% to a minimum of 83%. The accuracy of FWA also varied in the same way, with the lowest accuracy reaching 60%, and the lowest accuracy of RRT algorithm was also 60%. The minimum accuracy of LSTM was 50%, and the accuracy of PSO was even close to 40%. However, unlike other algorithms, when processing an initial sample size of 4500, the accuracy of IFWAM remains at 95%, which is higher than RRT’s 75% and PSO’s 61% accuracy. The accuracy rate rebounded with a decrease to 82.1% (F1 value = 85.3%), as the newly added samples contained a large number of low-risk scanning behaviors (false positive rate increased by 23%). The discrimination of grey area traffic by the reflection model relies on manual rule calibration, and in the future, a semi supervised learning mechanism needs to be integrated. At a sample size of 4500, the accuracy of IFWAM rebounded to 95%, but its actual performance needs to be analyzed in conjunction with F1 value: when the proportion of malicious samples decreased to 2.1%, the model accuracy remained at 91.3%, and the F1 value reached 87.5%, significantly higher than PSO (F1 = 42.1%) and RRT (F1 = 59.8%), verifying its stable recognition ability for minority classes.

To verify the comprehensive effectiveness of the proposed algorithm in a real network environment, empirical testing was conducted in an enterprise level network security monitoring scenario. This scenario includes a real hybrid traffic generator that continuously generates background traffic (web browsing, email, video conferencing) and injects various known and unknown attack traffic (such as DDoS flood attacks, port scanning, SQL injection, and simulated zero day attack traffic). All algorithms are deployed as a real-time analysis engine to extract features and identify malicious behavior from traffic, and record their processing efficiency and accuracy. The test lasted for 8 h with a total traffic of approximately 2 TB. In this simulated environment, the comprehensive defense performance of different algorithms was evaluated, and the results are shown in Table 6.

As shown in Table 6, the proposed IFWAM model demonstrates comprehensive advantages in simulated real-world application scenarios. Its high unknown attack detection rate (73.5%) and low false alarm rate (1.5%) are attributed to the strong generalization ability and anti-interference ability brought by multi-objective optimization and fuzzy neural networks, indicating its effectiveness in dealing with new threats. IFWAM reduces system response latency to 135 milliseconds and significantly reduces CPU usage, which is directly attributed to its efficient feature selection mechanism that compresses data dimensions from over 200 to 45 dimensions, greatly improving computational efficiency and proving its feasibility for deployment in high-throughput network environments. In contrast, PSO and standard FWA algorithms have shortcomings in both accuracy and efficiency; Although the RRT algorithm has low latency, its detection rate and false alarm rate indicators are difficult to meet actual security requirements; Although the LSTM model has a decent detection rate, its high computational resource consumption and high latency make it difficult to apply on resource constrained edge security devices.

Conclusion

The research aims to design an IFWAM that combines Gaussian operators and adaptive functions to improve sensitivity and accuracy in network data processing. During the process, the FS and FWA were optimized by changing the single-objective selection to a multi-objective multi label model. A fitness function and Gaussian operator were introduced into the FWA and fused with a fuzzy neural network to construct a new model. The design included technical analysis of multi-objective optimization models and how to evaluate the performance of datasets in long-term experimental environments. The experimental results showed that the multi-objective algorithm could collect data information up to 5000, while other algorithms could collect up to 3000. In the sensitivity experiment, the algorithm maintained a high sensitivity of 70%-100%. In terms of perceptual quality, the maximum difference in perceptual quality before and after applying multi-objectives was 0.15, and the minimum was 0.03. In the error experiment of the FWA, the maximum difference between the actual value and the predicted value was only 0.3, the lowest comprehensive performance evaluation value was 5, the adaptive adjustment time was reduced by half compared to before improvement, and the recognition accuracy of malicious behavior remained at a high level, ranging from 83% to 95%. The study addressed the

potential drawbacks of using Gaussian operators and adaptive functions in FWAs, which are mainly reflected in the problem of the algorithm being prone to getting stuck in local optima. The decline in accuracy observed on medium-sized datasets mainly results from the imbalance in the adaptation between model complexity and data scale, as well as the interference of nonlinear feature interaction. When the dimension of the feature space does not match the information volume of the medium dataset, some redundant parameters are prone to capture noise patterns. Multi-label correlation leads to pseudo-correlation of certain feature subsets in a limited number of samples. Although the adaptive weight adjustment mechanism and fuzzy rule-driven feature screening designed by the research method have effectively alleviated this issue, in the future, the feature subset can still be further streamlined by introducing sparse constraints, and the generalization ability of the lightweight model can be verified in edge computing scenarios. The research optimized the traditional fireworks algorithm by introducing Gaussian operators and adaptive functions, significantly enhancing the sensitivity and accuracy of network data processing. However, for more complex cyber threats (such as zero-day attacks), in the future, it is necessary to further enhance the model's predictive capabilities by integrating deep learning technology with big data behavior analysis frameworks. Specifically, the response strategy for zero-day attacks can be achieved by constructing a deep neural network based on spatio-temporal feature extraction and using deep learning to automatically learn the implicit patterns of attack behaviors from the traffic sequence. It can also integrate multi-source threat intelligence big data and establish a dynamically updated attack feature knowledge graph to enhance context awareness capabilities. There are still some generalization challenges in the research methods, and the observed decrease in accuracy on medium-sized datasets is mainly due to the imbalance between model complexity and data size adaptation, as well as the interference of nonlinear feature interactions; The sensitivity of multi label feature selection to sample correlation may affect stability in heterogeneous network environments.

### Data availability

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Received: 18 July 2025; Accepted: 6 November 2025

Published online: 19 December 2025

### References

- Hasanvand, M., Nooshyar, M., Moharamkhani, E. & Selyari, A. Machine Learning Methodology for Identifying Vehicles Using Image Processing, *AIA*, vol. 1, no. 3, pp. 170–178, Apr. (2023). <https://doi.org/10.47852/bonviewAIA3202833>
- Movassagh, A. A. et al. Artificial neural networks training algorithm integrating invasive weed optimization with differential evolutionary model, *J. Ambient Intell. Humaniz. Comput.*, pp. 1–9, Mar. (2023). <https://doi.org/10.1007/s12652-020-02623-6>
- Bhosle, K. & Musande, V. Evaluation of Deep Learning CNN Model for Recognition of Devanagari Digit, *Artif. Intell. Appl.*, vol. 1, no. 2, pp. 114–118, Feb. (2023). <https://doi.org/10.47852/bonviewAIA3202441>
- Alzubi, O. A. et al. An optimal pruning algorithm of classifier ensembles: dynamic programming approach. *Neural Comput. Appl.* **32**, 16091–16107. <https://doi.org/10.1007/s00521-020-04761-6> (Feb. 2020).
- Alzubi, O. A., Alzubi, J. A., Al-Zoubi, A. M., Hassonah, M. A. & Kose, U. An efficient malware detection approach with feature weighting based on Harris Hawks optimization, *Cluster Comput.*, pp. 1–19, Nov. (2022). <https://doi.org/10.1007/s10586-021-03459-1>
- Waqas, M. et al. The role of artificial intelligence and machine learning in wireless networks security: Principle, practice and challenges, *Artif. Intell. Rev.*, vol. 55, no. 7, pp. 5215–5261, Feb. (2022). <https://doi.org/10.1007/s10462-022-10143-2>
- Ping, H. Network information security data protection based on data encryption technology, *Wirel. Personal Commun.*, vol. 126, no. 3, pp. 2719–2729, Jun. (2022). <https://doi.org/10.1007/s11277-022-09838-0>
- Zhao, X. A network security algorithm using SVC and sliding window. *Wirel. Netw.* **29** (1), 345–351. <https://doi.org/10.1007/s11276-022-03064-z> (Sep. 2023).
- Hong, S., Seo, H. & Yoon, M. Data auditing for intelligent network security monitoring, *IEEE Commun. Mag.*, vol. 61, no. 3, pp. 74–79, Dec. (2022). <https://doi.org/10.1109/MCOM.003.2200046>
- Taheri, R., Ahmed, H. & Arslan, E. Deep learning for the security of software-defined networks: A review. *Cluster Comput.* **26** (5), 3089–3112. <https://doi.org/10.1007/s10586-023-04069-9> (Jul. 2023).
- Thakkar, A. & Lohiya, R. A survey on intrusion detection system: Feature selection, model, performance measures, application perspective, challenges, and future research directions, *Artif. Intell. Rev.*, vol. 55, no. 1, pp. 453–563, Jul. (2022). <https://doi.org/10.1007/s10462-021-10037-9>
- Rashid, M., Kamruzzaman, J., Imam, T., Wibowo, S. & Gordon, S. A tree-based stacking ensemble technique with feature selection for network intrusion detection. *Appl. Intell.* **52** (9), 9768–9781. <https://doi.org/10.1007/s10489-021-02968-1> (Jan. 2022).
- Sah, G., Banerjee, S. & Singh, S. Intrusion detection system over real-time data traffic using machine learning methods with feature selection approaches, *Int. J. Inf. Secur.*, vol. 22, no. 1, pp. 1–27, Oct. (2023). <https://doi.org/10.1007/s10207-022-00616-4>
- Hema, M. S., Maheshprabhu, R., Reddy, K. S., Guptha, M. N. & Pandimurugan, V. Prediction analysis for Parkinson disease using multiple feature selection & classification methods. *Multimed Tools Appl.* **82** (27), 42995–43012. <https://doi.org/10.1007/s11042-023-15280-6> (Apr. 2023).
- Mounica, B. & Lavanya, K. Feature selection with a deep learning based high-performance computing model for traffic flow analysis of Twitter data. *J. Supercomput.* **78**, 15107–15122. <https://doi.org/10.1007/s11227-022-04468-6> (Apr. 2022).
- Li, H. et al. An Optimization-Based path planning approach for autonomous vehicles using the DynEFWA-Artificial potential field. *IEEE T INTELL. Veh.* **7** (2), 263–272. <https://doi.org/10.1109/TIV.2021.3123341> (June 2022).
- Zhang, J. PSO-Based sparse source location in Large-Scale environments with a UAV swarm. *IEEE T INTELL. TRANSP.* **24** (5), 5249–5258. <https://doi.org/10.1109/TITS.2023.3237570> (May 2023).
- Guo, H., Li, J., Liu, J., Tian, N. & Kato, N. A Survey on Space-Air-Ground-Sea Integrated Network Security in 6G, in *IEEE Communications Surveys & Tutorials*, vol. 24, no. 1, pp. 53–87, Firstquarter (2022). <https://doi.org/10.1109/COMST.2021.3131332>
- Mahmud, S., Collings, W., Barchowsky, A., Javaid, A. Y. & Khanna, R. Global maximum power point tracking in dynamic partial shading conditions using ripple correlation control. *IEEE T IND. APPL.* **59** (2), 2030–2040. <https://doi.org/10.1109/TIA.2022.3228227> (March–April 2023).
- Yang, W. et al. Multi-Objective Optimization of High-Power Microwave Sources Based on Multi-Criteria Decision-Making and Multi-Objective Micro-Genetic Algorithm, *IEEE T ELECTRON. DEV.*, **70**, 7, 3892–3898, doi: <https://doi.org/10.1109/TED.2023.3280157> (2023).

21. Tian, Y., Su, X., Su, Y. & Zhang, X. EMODMI: A Multi-Objective Optimization Based Method to Identify Disease Modules, *IEEE T EMERG TOP COM* vol. 5, no. 4, pp. 570–582, Aug. (2021). <https://doi.org/10.1109/TETCI.2020.3014923>
22. Kim, S., Park, K. J. & Lu, C. A Survey on Network Security for Cyber-Physical Systems: From Threats to Resilient Design, *IEEE Communications Surveys & Tutorials*, vol. 24, no. 3, pp. 1534–1573, thirdquarter (2022). <https://doi.org/10.1109/COMST.2022.3187531>
23. Ma, L. Pareto-Wise ranking classifier for multiobjective evolutionary neural architecture search. *IEEE T EVOLUT COMPUT.* **28** (3), 570–581. <https://doi.org/10.1109/TEVC.2023.3314766> (June 2024).
24. Vellingiri, J., Vedhavathy, T. R., Senthil Pandi, S. & Bala Subramanian, C. Fuzzy logic and CPSO-optimized key management for secure communication in decentralized IoT networks: A lightweight solution. *Peer-to-Peer Netw. Appl.* **17** (5), 2979–2997. <https://doi.org/10.1007/s12083-024-01733-8> (June 2024).
25. Ming, F., Gong, W. & Gao, L. Adaptive Auxiliary Task Selection for Multitasking-Assisted Constrained Multi-Objective Optimization [Feature], *IEEE COMPUT. INTELL. M.* **18**, 2, 18–30, doi: <https://doi.org/10.1109/MCI.2023.3245719>. May (2023).
26. Liu, L., Zhang, Z., Chen, G. & Zhang, H. Resource management of heterogeneous cellular networks with hybrid energy supplies: A Multi-Objective optimization approach. *IEEE T WIREL. COMMUN.* **20** (7), 4392–4405. <https://doi.org/10.1109/TWC.2021.3058519> (July 2021).
27. Shi, Y. et al. Feb., Consensus Learning for Distributed Fuzzy Neural Network in Big Data Environment, *IEEE T EMERGING TOPICS COMPUTATIONALS INTELL.*, vol. 5, no. 1, pp. 29–41, (2021). <https://doi.org/10.1109/TETCI.2020.2998919>
28. Liu, F., Zhang, G. & Lu, J. Multisource Heterogeneous Unsupervised Domain Adaptation via Fuzzy Relation Neural Networks, *IEEE T FUZZY SYST* vol. 29, no. 11, pp. 3308–3322, Nov. (2021). <https://doi.org/10.1109/TFUZZ.2020.3018191>
29. Yashudas, A. et al. Deep-Cardio: recommendation system for cardiovascular disease prediction using IoT network. *IEEE Sens. J.* **24** (9), 14539–14547. <https://doi.org/10.1109/JSEN.2024.3373429> (March 2024).
30. Sahoo, S. K. et al. An arithmetic and geometric mean-based multi-objective moth-flame optimization algorithm. *Cluster Comput.* **27** (5), 6527–6561. <https://doi.org/10.1007/s10586-024-04301-0> (March 2024).
31. Chatur, N., Bose, T. & Adhya, A. Planning cost-efficient FiWi access network with joint deployment of FWA and FTTH. *IEEE Trans. Commun.* **72** (9), 5688–5703. <https://doi.org/10.1109/TCOMM.2024.3384933> (April 2024).

## Author contributions

L.Z. and C.L. processed the numerical attribute linear programming of communication big data, and the mutual information feature quantity of communication big data numerical attribute was extracted by the cloud extended distributed feature fitting method. L.T. and J.W. combined with fuzzy C-means clustering and linear regression analysis, the statistical analysis of big data numerical attribute feature information was carried out, and the associated attribute sample set of communication big data numerical attribute cloud grid distribution was constructed. C.L. and Y. X. did the experiments, recorded data, and created manuscripts. All authors read and approved the final manuscript.

## Funding

No funding received.

## Declarations

## Competing interests

The authors declare no competing interests.

## Additional information

**Correspondence** and requests for materials should be addressed to X.Y.

**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Open Access** This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025