ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 2 - 2026

Anomalous Data Evaluation of Power IoT Network Based on Machine Learning Situational Awareness Modeling

Guobang Ban[1,*], *Yumin He*[1], *Guanghui Xi*[1], *Xinbiao Xiong*[1], *Jiangang Liu*[1] *and Siqi Guo*[1]

[1] Guizhou Power Grid Co., Ltd. Electric Power Research Institute, Guiyang, Guizhou, 550000, China

SUMMARY: *With the diversification of the means of network attacks, network security has been increasingly emphasized. This paper carries out an in-depth study of power IoT network security situational awareness, anomalous data assessment and machine learning related technologies, proposes a GA-LightGBM network security situational awareness method based on PRF-RFECV feature optimization, constructs an anomalous data prediction model for power IoT network, IPSO-BiLSTM, and conducts experimental validation of the model effectiveness. The results show that the mean square error MSE is reduced from 0.0033 to 0.0024, the coefficient of determination R^2 is increased from 0.8943 to 0.9232, and the accuracy rate is increased by 3.8 percentage points compared with the unoptimized LightGBM posture assessment model, proving that the accuracy rate based on the PRF-RFECV-GA-LightGBM network security situational awareness methodology IPSO-BiLSTM. RFECV-GA-LightGBM network security posture assessment model has high accuracy and low error. Meanwhile, compared with other benchmark models, the results predicted using the IPSO-BiLSTM model have higher accuracy and lower error rate, which verifies that the model proposed in this paper is more applicable to the increasingly complex network environment.*

KEYWORDS: *machine learning; power IoT network; situational awareness; anomaly data prediction; PRF-RFECV-GA-LightGBM; IPSO-BiLSTM*

1 Introduction

The Internet of Things (IOT) is the practical application of IOT technology in the electric power system, and the concept was first proposed by the State Grid [1]. It makes full use of modern information technology, advanced communication technology such as “big cloud, material, mobile and intelligent chain” to realize the interconnection of all things and human-machine interaction in each link of the electric power system, and to improve the ability of automatic data collection, automatic acquisition and flexible application. Typical IOT network architecture includes three layers: perception layer, transmission layer, and application layer. The perception layer is to acquire raw data from the physical world with the help of IOT devices and process it simply; the transmission layer is responsible for data transmission and communication through various communication networks, such as wired or wireless communication; the application layer processes and analyzes the received data intelligently and in depth to provide decision-making support [2, 3]. Data is the foundation of power business and the driving force of power business operation. If you want the power

*Banguobang@126.com
https://doi.org/10.65102/is2026845 Published: 30 April, 2026

business to provide stable and high-quality services, you need to ensure that the data is correct, that is, the integrity of the collected data should be ensured. However, IOT terminals are deployed in complex and widely distributed environments, which are easy to be directly damaged or have attack devices installed on them [4, 5]. Access device resources are limited, and if effective access identity authentication is not possible, it may lead to attackers pretending to be legitimate devices to access the network, and then stealing or tampering information [6, 7]. Terminal resources are limited, and it is impossible to protect data with high strength encryption, which can easily lead to data leakage [8]. Attackers utilize IOT communication protocol vulnerabilities to launch replay attacks, false data injection, and other attacks on the network, leading to abnormal data transmission, loss, and other abnormalities generated [9, 10]. This shows that assessing abnormal data is necessary to ensure the stable operation of power IOT.

In the early days, rule-based or statistical methods were often used for network data anomaly detection and assessment in the electric power IOT environment, which not only made it difficult to cover a comprehensive range of anomalies, but also made the results of anomalous data assessment inaccurate and not real-time in complex environments or when the data quality was low [11-14]. In addition, the data in the electric power IoT environment presents multimodal, such as text, signal, image and other types of data, and the above methods are difficult to deal with multimodal data. With the development of artificial intelligence technology, machine learning (ML) models are widely used in the field of anomaly detection and prediction.The basic principle of ML anomaly detection is to learn the pattern of normal data and then determine whether the new data is anomalous or not according to the degree of deviation from the normal pattern.

Literature [15] designed an unsupervised machine learning based detection scheme for detecting hidden data integrity attacks faced by smart grids, which focuses on feature extraction with the help of Isolated Forest algorithm and Principal Component Analysis, which effectively improves the accuracy of identifying attacks in unlabeled data. Literature [16] developed an ML-based active anomaly detection framework, which focuses on early detection and identification using algorithms such as Isolated Forest, thus effectively capturing subtle deviations before smart grid cyber-attacks and realizing early warning. Literature [17] focuses on grid security and creates an unsupervised anomaly detection algorithm that combines Gaussian process and one-class support vector machine for false data injection attacks in the grid, thus effectively identifying tampering and improving microgrid protection. Whereas, literature [18] utilizes spatio-temporal correlation to identify tampering and cleans the data by denoising the self-encoder to detect false data injection attacks in IoT environments, which significantly outperforms the support vector machine scheme. Literature [19] utilizes ML, especially gated loop units, and combines it with hierarchical federated training to enhance detection and privacy protection, thus effectively improving the resilience of IoT networks facing distributed denial of service. Literature [20] combines multiple deep learning models with named entity recognition to efficiently identify and protect power IoT structured and unstructured sensitive data, reducing the risk of data leakage. Literature [21] utilizes support vector machines and long and short-term memory networks to predict and mitigate information loss, thereby significantly improving the accuracy and reliability of network management to cope with data loss problems caused by complex data flows in IoT networks. Literature [22] collects data from home appliances via IoT sensors and performs device anomaly detection and energy consumption prediction via ML, in which Prophet and LightGBM models outperform vector autoregressive models in identifying point anomalies and predicting energy consumption, thus enhancing power system maintenance. Literature

[23] proposes a parallel random forest-based anomaly prediction algorithm for the electric power IoT environment, which extracts key data features and optimizes the parameters, thus achieving more accurate anomaly prediction and data encryption screening of electric power big data. It can be seen that the ML model provides support for data anomaly detection and evaluation in the electric power IOT environment from the aspects of anomaly prediction, anomaly detection, and attack risk identification.

In addition, situational awareness is the process of acquiring various elements in the environment, understanding them and predicting their future states from the perspective of time and space. Literature [24] constructed a multi-scale situational awareness model, which is based on FARIMA dynamic thresholds for anomaly detection, thus realizing an effective quantitative assessment of traffic anomalies and high-risk protocols in substation communication networks. Literature [25] introduces a security situational awareness model that establishes data correlation through time-series analysis and performs multilevel anomaly determination from the data layer, the feature layer to the decision-making layer, thus effectively improving the detection accuracy and control capability of abnormal behavior of power monitoring data interaction. Literature [26] establishes a distributed situational awareness architecture model, realizes event collection, hybrid processing and dynamic response through modular components, and verifies the feasibility and scalability of deployment in IoT production scenarios such as information security. Literature [27] combines the ML framework of linear discriminant analysis and radial-based neural networks to process attack data for reliable assessment and high-precision prediction of power information network security situational awareness. Literature [28] systematically reviews ML and deep learning methods for the anomaly detection problem in network situational awareness, and proposes an integrated model that incorporates multimodal data fusion, online adaptive detection, and federated learning, thus providing an innovative framework for constructing an efficient, interpretable, and privacy-preserving detection system. The above study shows that the situational awareness model is capable of comprehensively analyzing and predicting power and IoT security, and can meet the real-time demand and anomaly assessment of power IoT.

In this paper, based on the machine learning method, research is carried out on the problem of network security situational awareness and anomalous data assessment of power IoT. Firstly, based on the PRF-RFECV algorithm for feature optimization and the GA algorithm for global optimization of important hyperparameters of LightGBM, the network security situational awareness model PRF-RFECV-GA-LightGBM is constructed, and the advantages of the benchmark algorithm LightGBM over other algorithms are verified, and the final model is compared with GA-LightGBM, LightGBM for performance comparison. Secondly, the particle swarm algorithm is improved by combining it with BiLSTM to propose a network anomaly data prediction model based on IPSO-BiLSTM, and the effectiveness of the model is also verified.

2 Machine Learning Based Security Situational Awareness for Power IOT Networks

2.1 Power IoT Network Security Situational Awareness Modeling Framework

In this chapter, a GA-LightGBM approach for security situational awareness in power IoT networks based on PRF-RFECV feature preferences is proposed.The framework of the

PRF-RFECV-GA-LightGBM situational awareness model is shown in Fig. 1, which consists of three modules, namely, the Situation Extraction Module (SEXM), the Situation Assessment Module (SASM), and the Situation Visualization Module (SVIM) The framework is composed of three modules.

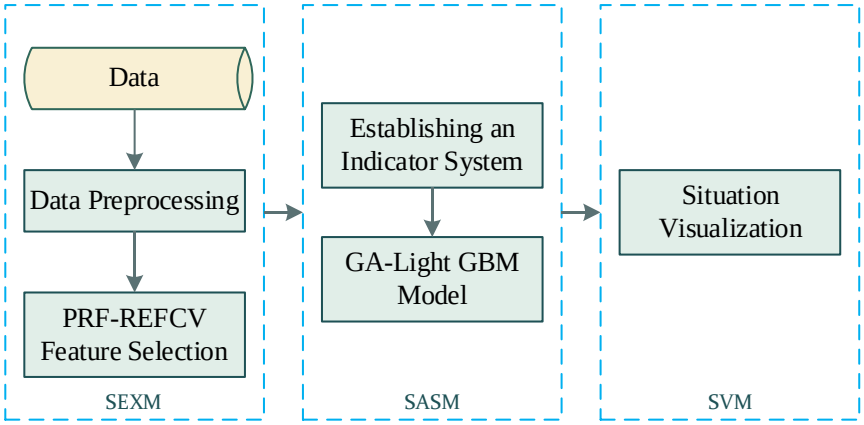

Figure 1: Framework of PRF-RFECV-GA-LightGBM situation awareness model

2.2 Network Security Situational Awareness Related Technologies

2.2.1 LightGBM Algorithm

LightGBM is an efficient gradient boosting framework for solving supervised learning problems.The core idea is to build a powerful integrated model using gradient boosting based decision trees.The main features of the LightGBM algorithm are gradient-based one-sided sampling (GOSS) decision tree algorithms, exclusive feature bundling (EFB), histograms, and leaf-growth strategies with depth restrictions (Leaf-wise).

Sample data $X=\left\{\left(x_i,y_i\right)\right\}_{i=1}^{n}$, LightGBM algorithm obtains the corresponding predicted values $\left\{\hat{y}_1,\hat{y}_2,\cdots,\hat{y}_i,\cdots,\hat{y}_n\right\}$, and the prediction is computed as:

$$\hat{y}_i=\sum_{k=1}^{K}f_k(x_i)=\hat{y}_i^{(k-1)}+f_k(x_i) \tag{1}$$

where $f_k(x_i)$ denotes the k th trained decision tree prediction and x_i is the input feature vector.

The corresponding regularization term is:

$$\Omega(f_t)=\gamma T+\frac{1}{2}\lambda\sum\nolimits_{j=1}^{T}w_j^2 \tag{2}$$

where T is the number of leaves in the tree, ω is the leaf weights, and γ and λ are parameters.

A regularization strategy is used to reduce overfitting during LightGBM learning:

$$F_t(x)=F_{t-1}(x)+\nu f_t(x) \tag{3}$$

where $0<v<1$ is the learning rate to update the model, the smaller the learning rate, the more regression trees need to be generated, the accuracy will be higher, but at the same time will increase the training time.

This leads to the formula for minimizing the objective function of the LightGBM algorithm:

$$\begin{aligned}\Theta_{\min} &= \arg\min\left\{\sum_{i=1}^{N} l\left(y_i,\hat{y}_i\right)+\Omega(f_t)+C\right\}\\ &= \arg\min\left\{\sum_{i=1}^{N} l\left(y_i,\hat{y}_i^{(t-1)}+f_t(x_i)\right)+\Omega(f_t)+C\right\}\end{aligned} \tag{4}$$

where $\hat{y}_i^{(t-1)}$ denotes the predicted value of the $t-1$th iteration, $\Theta_{\min}$ is the optimization parameter, C is a constant term, and $\sum_{i=1}^{N} l\left(y_i,\hat{y}_i\right)+\Omega(f_t)+C$ is the objective function of the algorithm.

Taylor's second order expansion of the objective function, the gain function of the leaf node can be reduced to:

$$Gain=\frac{1}{2}\left[\frac{G_L^2}{H_L+\lambda}+\frac{G_R^2}{H_R+\lambda}-\frac{\left(G_L+G_R\right)^2}{H_L+H_R+\lambda}\right]-\gamma \tag{5}$$

where H and G denote the first and second order derivatives of $\hat{y}_i^{(t-1)}$ in the objective function, i.e., respectively:

$$H_j=\sum_i \frac{\partial l(y_i,\hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)}} \tag{6}$$

$$G_j=\sum_i \frac{\partial^2 l(y_i,\hat{y}_i^{(t-1)})}{\partial (\hat{y}_i^{(t-1)})^2} \tag{7}$$

$\frac{G_L^2}{H_L+\lambda}$ denotes the post-split left subtree score, $\frac{G_R^2}{H_R+\lambda}$ is the right subtree score, and $\frac{(G_L+G_R)^2}{H_L+H_R+\lambda}$ is the node score before splitting, γ increases the the complexity cost of the nodes.

The grow-by-leaf strategy of the LightGBM algorithm, i.e., the leaf node with the largest splitting gain is selected to fit the best prediction $\hat{y}_i$.

2.2.2 Random Forest Algorithm

Random Forest (RF) belongs to the Bagging type of integrated algorithms, which is learned by sampling repeatedly with putback and incorporates the extraction of random features to form mutually independent training subsets to improve the randomness of generating decision trees.

Let the total number of samples N, there are B features, RF is based on the Gini index to calculate the importance of the features formula is:

$$GI_m(p)=\sum_{k=1}^{K}p_{mk}(1-p_{mk})=1-\sum_{k=1}^{K}p_{mk}^2 \tag{8}$$

where k represents a total of K categories and p_{mk} represents the weight of k in node m.

The importance of feature X_j in node m, i.e., the amount of change in the Gini index before and after the branching of node m is:

$$VIM_{jm}^{gini}=GI_m-GI_l-GI_r \tag{9}$$

where GI_l and GI_r represent the Gini indices of the new nodes on the left and right after branching.

If the node where the feature X_j appears in the decision tree i is in the set M, then the importance of the feature X_j in the first i decision tree is:

$$VIM_{ij}^{gini}=\sum_{m\in M}VIM_{jm}^{gini} \tag{10}$$

Assuming that there are a total of n decision trees in the random forest, the feature importance of feature X_j is:

$$VIM_{j}^{gini}=\sum_{i=1}^{n}VIM_{ij}^{gini} \tag{11}$$

Finally, all obtained importance scores were normalized:

$$VIM_j=\frac{VIM_j}{\sum_{i=1}^{B}VIM_i} \tag{12}$$

Parallel Random Forest (PRF) can utilize the characteristics of Random Forest, such as each decision tree is calculated separately, to construct a parallel random forest, so that multiple decision trees are calculated in parallel to learn, in order to avoid the training inefficiency caused by the construction of multiple decision trees. The principle is that the master divides the data to be processed into multiple data blocks and sends them together with their corresponding execution programs to each process in the process pool. After the parallel computation is completed, the master summarizes each result and organizes the results in the format required by the user.

2.2.3 Feature Optimization Method

Recursive feature elimination by cross-validation (RFECV) is commonly used in feature selection problems.RFE can be described as iterating over the base model for feature importance, each iteration eliminates the features with the lowest importance until the feature set is a specified number, and the final order of feature importance is the order of elimination.RFECV is a process of cross-validation on top of RFE for different combinations

of features, and the base class model to calculate the validation error of all subsets and select the subset with the smallest error rate as the optimal feature subset.

The PRF-RFECV algorithm is a combined algorithm that combines PRF and RFECV, and its specific process is as follows:

(1) Construct multiple decision trees using PRF on the preprocessed dataset, and sort each feature from largest to smallest according to the Gini index.

(2) Delete one or more features with the smallest feature importance one by one until the number of feature subsets is one, and record the feature subsets after each iteration in turn, and take them as new feature subsets.

(3) All feature subsets are cross-validated based on the base model with 10 folds to get the highest rated feature subset as the best feature set.

2.2.4 Genetic algorithms

Genetic Algorithm (GA) is an optimization algorithm inspired by evolutionary processes in nature for solving search and optimization problems. It simulates evolutionary processes such as natural selection, crossover and mutation to find the optimal or near-optimal solution to a problem by continuously evolving individuals in a population. The genetic algorithm flow is shown in Figure 2.

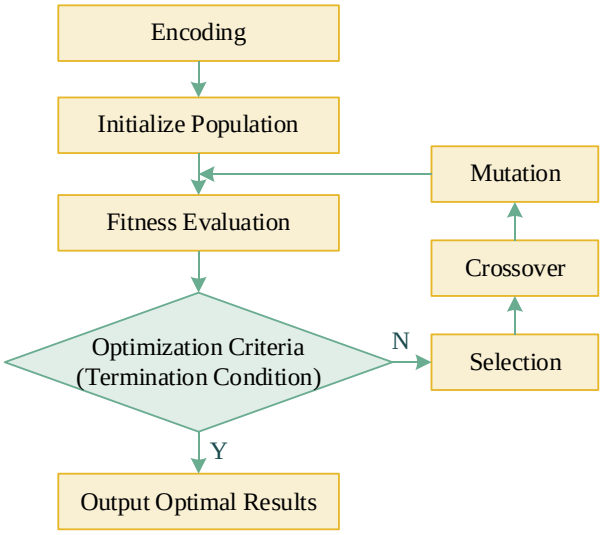

Figure 2: Genetic Algorithm process

The main process characteristics of GA are as follows:

(1) Encoding

Representation of the solution of the problem in the form of a chromosome or genome to enable genetic operations such as selection, crossover and mutation to be performed in the framework of a genetic algorithm.

(2) Initializing population

The initialization population is the starting point of the GA search problem, which is a group of randomly generated individuals that all represent potential solutions to the optimization problem.

(3) Individual fitness evaluation

In this process, the fitness function is used to evaluate the superiority and performance of each chromosome. Individuals with higher fitness scores represent better solutions, which are more likely to be selected for reproduction and whose traits will be expressed in the next

generation, and through the fitness evaluation process, only the top performing solutions are retained for further replication processes.

(4) Genetic operations

In binary genetic operations, there are three main operations: selection, crossover, and mutation.

1) Selection

Selection is the component that guides the algorithm to solve the problem by choosing individuals with high fitness over those with low fitness, the probability of selecting a particular individual is proportional to its fitness. The probability of selecting an individual is:

$$P(b_j) = \frac{f(b_j)}{\sum_{k=1}^{n} f(b_k)} \tag{13}$$

where $f(b_j)$ is the individual b_j fitness value and $\sum_{k=1}^{n} f(b_k)$ is the sum of all individual fitness values.

2) Crossover

Selection in which some of the chromosomes of a sample of both parents are interchanged to create two new chromosomes representing the offspring.

3) Mutation

In order to avoid the genetic algorithm from falling into a local optimum solution during the optimization process, it is necessary to mutate individuals during the search process. The purpose of the mutation operation is to periodically and randomly update the population, introduce diversity and novelty into the chromosomes, and guide the algorithm to search in unknown regions of the solution space.

(5) Termination conditions

The newly generated offspring chromosomes undergo selection, crossover, and mutation processes to calculate their fitness and verify the termination conditions. Two commonly used termination conditions, one is that the maximum number of generations has been reached and the genetic algorithm ends. The second is that there is no significant improvement of the individual in the past generations, i.e., the difference between the current optimum and the optimum obtained a few generations before the predetermined number of generations is made, and the algorithm can be stopped if the difference is less than a certain threshold value. If the termination conditions are not met, the selection, crossover and mutation processes are repeated to produce superior chromosomes with higher performance.

2.3 Cybersecurity Situational Awareness Assessment Models

2.3.1 Optimization of LightGBM

In the posture assessment module, in order to improve the accuracy and generalization ability of the model and avoid overfitting, the genetic algorithm is used to obtain the optimal learning rate, maximum tree depth and the number of base tree trees for the LightGBM model.

Firstly, the potential solution of the problem is encoded in floating point numbers according to Eqs. (2) to (3), and the mapping relation is:

$$X = \left(x_1, x_2, \ldots, x_n\right)^T \Rightarrow V = \left(v_1, v_2, \ldots, v_m\right)^T \tag{14}$$

where X represents the decision vector, V represents the chromosome, the number of genes m is equal to the number of decision variables n, and the maximum number of evolutionary generations is set to 300.

The Mean Absolute Percentage Error (MAPE) is selected as the fitness function to measure the goodness of the prediction results, and the formula is as follows:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|y_i - \hat{y}_i\right|}{y_i} \tag{15}$$

where y_i is the true value and $\hat{y}_i$ is the predicted value, in order to satisfy the requirements of non-negativity, continuity and maximization, the fitness function can be finally converted into:

$$F(y) = \frac{1}{1+M(y)} \tag{16}$$

In order to avoid falling into the local optimum point, a generalized and efficient tournament selection algorithm is chosen, which repeatedly selects the best individuals into the offspring population based on fitness, so that the resulting individuals constitute the new generation population.

The simulated binary crossover is selected as the crossover operator, and the boundary variation is the variation operator. In order to avoid the problem of too small probability, which leads to the premature convergence of the genetic process, let the crossover and mutation probabilities be P_c and P_m, respectively, with the restriction ranges $\left[P_{c\min}, P_{c\max}\right]$ and $\left[P_{m\min}, P_{m\max}\right]$, where $P_{c\min} = 0$, $P_{c\max} = 0.9$, $P_{m\min} = 0.01$ and $P_{m\max} = 0.1$. Setting the population's all-individual fitness mean to f_{avg} and the fitness of crossover and variation to f' and f, respectively, yields:

$$P_c = \begin{cases} P_{c\max} - \dfrac{\left(P_{c\max} - P_{c\min}\right)\left(f_{\max} - f'\right)}{f_{\max} - f_{avg}}, & f' \geq f_{avg} \\ P_{c\max}, & f' < f_{avg} \end{cases} \tag{17}$$

$$P_m = \begin{cases} P_{m\max} - \dfrac{\left(P_{m\max} - P_{m\min}\right)\left(f_{\max} - f'\right)}{f_{\max} - f_{avg}}, & f \geq f_{avg} \\ P_{m\max}, & f < f_{avg} \end{cases} \tag{18}$$

As the genetic process continues to evolve, populations representing new solution sets are generated to better adapt to the environment, eventually reaching an arbitrary threshold that is preset and stopping the iteration to obtain the optimal hyperparameters for the LightGBM model.

2.3.2 Assessment process

The main flow of the power interconnection network situational awareness model PRF-RFECV-GA-LightGBM proposed in this paper is shown in Fig. 3. In the posture

assessment step designed in this paper, the stopping conditions of the algorithm are set to the assessment accuracy and the maximum number of iterations 300, and if either condition is reached, the procedure is terminated and the network security posture awareness model of GA-LightGBM is generated, which is followed by the input of the samples optimized by the features of the PRF-RFECV algorithm, and finally the posture value is derived according to the constructed assessment index system.

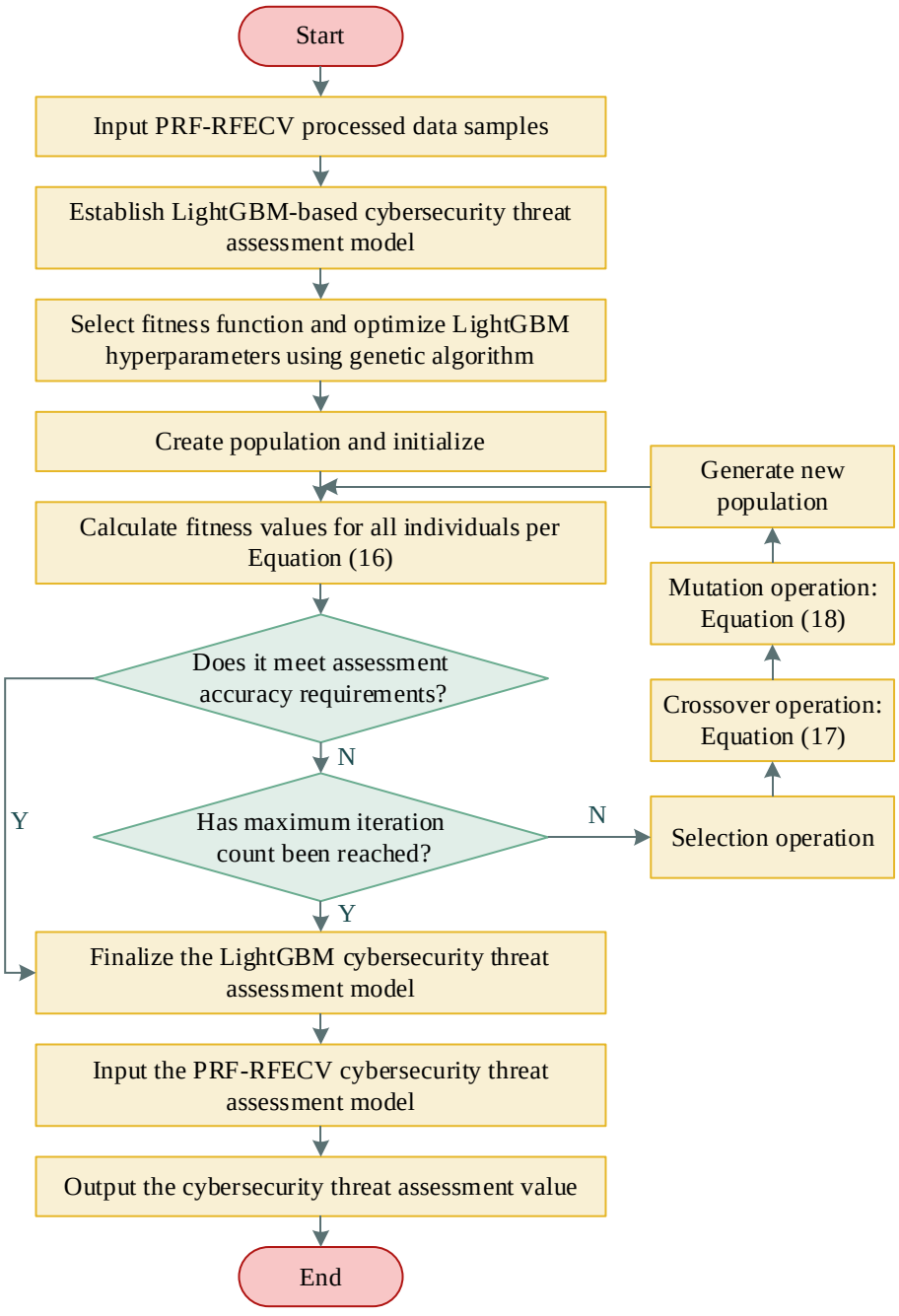

Figure 3: LightGBM situation awareness model optimized based on GA

2.4 Network Security Posture Assessment Indicator System Construction

2.4.1 Quantification and weighting of indicators

The weight of the posture indicators is the key to establish the indicator system, and this paper

adopts the expert empirical method to assign weights to the indicators.

In order to build a complete indicator system, it is necessary to quantitatively operate the security elements in the network environment and convert them into numerical type. This paper combines the Common Vulnerability Scoring System (CVSS) as the basis to do quantitative processing of the collected data, some of which are formulated as follows:

(1) Vulnerability degree:

$$c = \frac{\sum_{k=1}^{n} \sum_{l=1}^{S} w_{kl} N_{kl} I_k}{C_n} \tag{19}$$

where C is the level of vulnerability hazard, C_n is the total number of vulnerabilities present in the network, n is the number of devices, S is the type of vulnerability present in the network, w_{kl} is the weight of the vulnerability, and N_{kl} is the number of k devices containing l vulnerabilities. I_k is the importance of k devices in the network, calculated as:

$$I_k = \frac{D_t}{\sum_{i=1}^{n} D_i} \tag{20}$$

where n is the number of devices and D_i is the degree of importance of the data saved in i devices, the degree takes the following values:

$$I_i = \begin{cases} 1.0, & \text{Extremely important} \\ 0.8, & \text{Very important} \\ 0.5, & \text{Generally important} \end{cases} \tag{21}$$

(2) Network topology:

$$topo = \sum_{k=1}^{n} TP_k \tag{22}$$

$$TP_i = \begin{cases} 1.0, & n \in [0,4] \\ 0.6, & n \in (4,6] \\ 0.2, & n \in (6,+\infty) \end{cases} \tag{23}$$

where $topo$ is the quantized value of the topology in the network environment, TP_k is a specific score metric for the topology used in the k network, and n is the number of nodes in the network environment being measured.

(3) Attack severity:

$$A = \frac{\sum_{k=1}^{n} \sum_{l=1}^{S} r_{kl} b_k Q_{kl}}{a_n} \tag{24}$$

where A is the level of damage after suffering an attack, n is the total number of devices,

S is the number of types of attacks suffered, r_{kl} is the level of k devices when they suffered an l attack, b_k is the level of vulnerability on k devices, Q_{kl} is the number of times k devices suffered an l type of attack in a period of time, and a_n is the period of time number of types of attacks in a period of time.

(4) Degree of occurrence of attacks:

$$f = \frac{N}{t} \tag{25}$$

where f is the frequency of security alert events, t is a fixed time period, and N is the total number of security alert events occurring in the t time period.

(5) The degree of packet impact:

$$Y_k = \frac{b_k}{\sum_{k=1}^{n} b_k} \tag{26}$$

where Y_k is the score of the degree of impact of packets on the network security posture, b_k is the number of k types of packets, and n is the total number of packet types.

(6) Traffic change rate:

$$l = \frac{L_{t-1}}{L_t} \tag{27}$$

where l is the degree of flow change, L_t is the amount of flow change in the current t time period, and L_{t-1} is the amount of flow change in the previous t time period.

2.4.2 Cybersecurity posture leveling

Through research and analysis, this paper normalizes the security posture values of the entire electric power interconnection Internet system to specified intervals as a way to classify the network system posture security into five levels: safe, low risk, medium risk, high risk, and ultra-high risk, with the corresponding dividing intervals of 0, (0,0.20], (0.20,0.50], (0.50,0.80], and (0.80, 1.00].

2.5 Experimentation and Analysis

2.5.1 Experimental data set and pre-processing

(1) Experimental dataset

In this paper, a set of publicly available NSL-KDD dataset is selected as the research object to verify the effectiveness of the PRF-RFECV-GA-LightGBM situational awareness model.The NSL-KDD dataset includes normal network traffic as well as some common cyber-attacks such as denial-of-service attack (DOS), Probe attack, user-to-root attack (U2R), remote user attack (R²L), etc.

(2) Experimental Data Preprocessing

1) Numericalization on the

NSL-KDD dataset, many columns belong to discrete data, so they must be represented numerically first. For example, in Protocol type, there are three protocol types, namely TCP,

UDP and ICMP, which are replaced by 1, 2 and 3 numbers respectively, thus completing the numericization of Protocol type.

2) Normalization

NSL-KDD data after the numerical value, each column of the magnitude of the difference is large, in order to unify the magnitude, to avoid large deviations, it is necessary to carry out the normalization process. The more commonly used methods are min-max normalization, standard deviation normalization and zero mean normalization, where zero mean normalization is used and the formula is shown below:

$$x' = \frac{X - \mu}{\sigma} \tag{28}$$

2.5.2 Experimental evaluation indicators

For a variety of machine learning algorithms, several classical algorithms are firstly selected to be applied to the field of situational assessment, and four algorithms, namely SVM, DNN, XGBoost and LightGBM, are selected for this topic, and the situational assessment results of the dataset adopted in this topic are analyzed through the four algorithms, and the comparisons are made in terms of the assessment of the situational value error, the R^2 coefficient of determination, and the accuracy of situational level assessment, etc. The optimal algorithm is selected and then improved and optimized. The optimal algorithm is selected and then improved and optimized.

(1) Mean square error (MSE):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2 \tag{29}$$

(2) Root Mean Square Error (RMSE):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2} \tag{30}$$

(3) Mean Absolute Error (MAE):

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{31}$$

(4) Discriminant coefficient (R^2):

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2} \tag{32}$$

where f is the fitted value, y is the true value, the numerator is the sum of the squares of the difference between the predicted data and the labeled mean, while the denominator is the square of the mean difference between the true value and the sample. The denominator is the square of the difference between the true value and the sample mean. R^2 is the sample-to-sample fit, and the closer the relationship between the two, the better, i.e., closer to 1 is best.

2.5.3 Experimental environment

(1) Configuration of experimental hardware environment

Since this experiment is a simulation experiment, this experiment is carried out on a personal laptop with Intel (R) Core(TM) i7-13700H processor with 2.3GHz and 64G RAM hardware environment.

(2) Configuration of experimental software environment

The main software environments used in this experiment are as follows: python 3.13, numpy 2.2, pandas 2.2, matplotlib 3.8.2.

2.5.4 Basic posture assessment algorithm selection

The partial sample results of SVM, DNN, XGBoost, and LightGBM for evaluating the test set of the NSL-KDD dataset are shown in Table 1. It can be seen that compared with the three algorithms of SVM, DNN, and XGBoost, LightGBM is more similar to the real value and real grade both in terms of the evaluated posture value and the corresponding posture grade.

Table 1: Partial evaluation results of the four algorithms

Sample number	Evaluation result				True level (true value)
	SVM	DNN	XGBoost	LightGBM	
2	Low risk (0.012)	Low risk (0.11)	Low risk (0.11)	Low risk (0.12)	Low risk (0.12)
24	Ultra-high risk (0.89)	Low risk (0.17)	Ultra-high risk (0.92)	Ultra-high risk (0.92)	Ultra-high risk (0.93)
45	Moderate risk (0.25)	Low risk (0.10)	Moderate risk (0.32)	High risk (0.76)	High risk (0.78)
90	Low risk (0.16)	Low risk (0.09)	Moderate risk (0.31)	Moderate risk (0.34)	Moderate risk (0.35)
108	Moderate risk (0.78)	Moderate risk (0.23)	Ultra-high risk (0.85)	Ultra-high risk (0.88)	Ultra-high risk (0.87)

(1) Comparison of evaluation indexes of algorithms

The various evaluation indexes of the four algorithms DNN, SVM, XGBoost and LightGBM are shown in Fig. 4. It can be seen that among the four algorithms, DNN has the largest error, followed by SVM and XGBoost, and the LightGBM algorithm has the lowest error, and similarly the R^2 value of LightGBM, 0.8943, is also the highest among the four algorithms, which indicates that the training effect of the LightGBM model is the best among the four models.

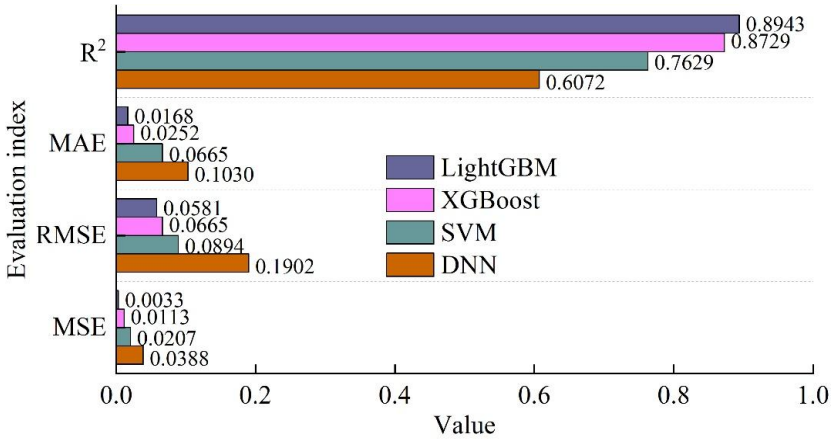

Figure 4: Basic algorithm evaluation indicators

(2) Comparison of accuracy rate for posture level assessment

The accuracy comparison of the four algorithms DNN, SVM, XGBoost, and LightGBM for the posture level is shown in Figure 5. It can be seen that DNN has the lowest accuracy rate for situational grade assessment, XGBoost is higher than SVM and DNN, and LightGBM has the highest accuracy rate among the four, reaching 0.9415.

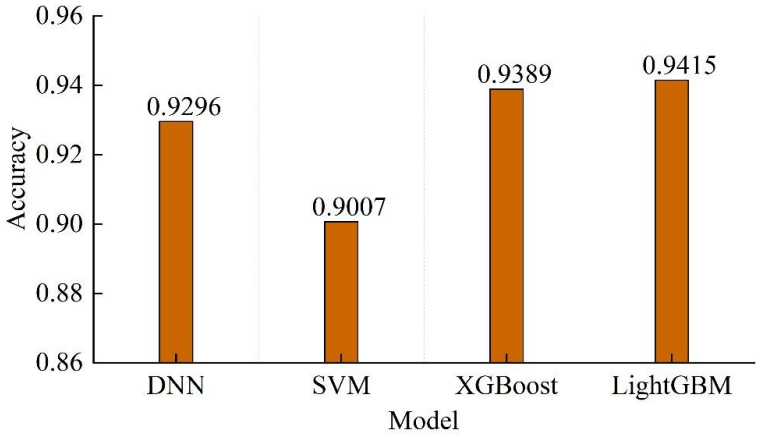

Figure 5: Accuracy rate of situation level assessment

Combining the above three results, it can be seen that LightGBM is ahead of the other three algorithms among the four algorithms of DNN, SVM, XGBoost, and LightGBM, no matter in the evaluation indexes of the algorithms or in the accuracy of the evaluation of the situational level. Therefore, LightGBM is chosen as the basic algorithm for subsequent improvement and optimization.

2.5.5 Comparison of optimization algorithm posture assessment results

(1) Combining the optimization results of PRF-RFECV and GA

The feature optimization is based on PRF-RFECV algorithm first, and then genetic algorithm (GA) is applied to improve and optimize the hyperparameters of LightGBM algorithm, and the optimization search process and MSE error after several iterations are shown in Fig. 6. It can be seen that the initial model error is nearly 0.00396, which is finally reduced to 0.00238 after several iterations.

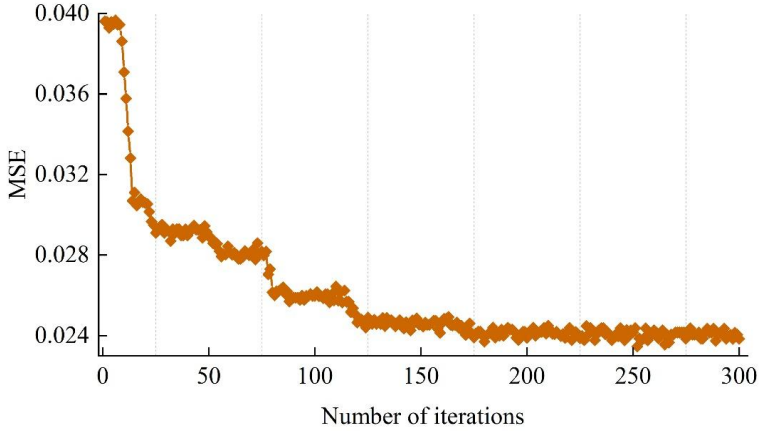

Figure 6: Optimization process and MSE variation curve

The optimal parameters when the error is lowest are brought into the LightGBM model to obtain the optimal power IoT network security situational awareness model PRF-RFECV-GA-LightGBM, and the final situational assessment results are shown in Figure 7.

It can be seen that the evaluation results of the LightGBM model optimized by PRF-RFECV and GA are basically consistent with the real values, except for a small number of samples such as sample No. 87, sample No. 262, sample No. 387, etc., which have some differences with the real values, the rest are not much different from the real values, and the corresponding situational grades of the assessed individual samples are basically the same as the real grades.

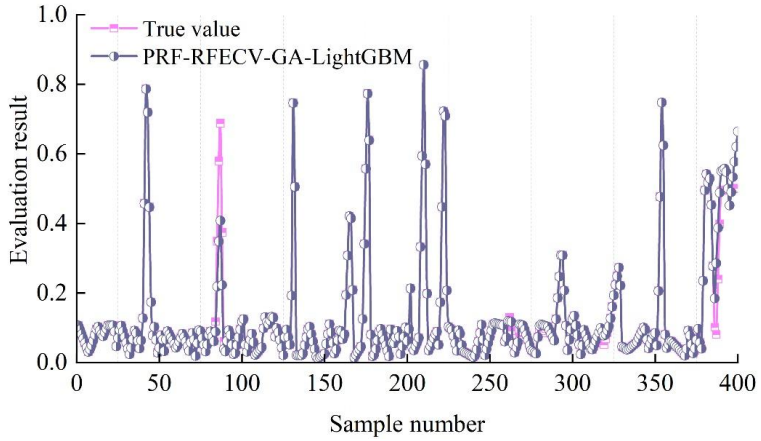

Figure 7: Evaluation results of PRF-RFECV-GA-LightGBM

(2) Comparison of evaluation results

The results of the unoptimized LightGBM, LightGBM optimized by genetic algorithm (GA-LightGBM) and PRF-RFECV-GA-LightGBM evaluated on the same dataset are compared, and some samples are randomly selected as shown in Table 2. It can be seen that the evaluation results of the three algorithms of LightGBM, GA-LightGBM, and PRF-RFECV-GA-LightGBM for sample No. 134 are not much different from the true value, and the posture grade and the true grade are all the same.175, 284 samples of GA-LightGBM, and PRF-RFECV-GA-LightGBM evaluated the The posture grades are all the same as the true grades, and the unoptimized LightGBM algorithm evaluates a large gap between the posture values and the true values.178, 315 samples only PRF-RFECV-GA-LightGBM evaluates the same posture grades as the true grades, and the other two evaluated posture values have a

considerable error between the true values and the true values. In summary, PRF-RFECV-GA-LightGBM is more similar to the unoptimized LightGBM and GA-LightGBM algorithms than the two algorithms, both in terms of the assessed posture values and the corresponding posture grades with the true values and true grades.

Table 2: Partial evaluation results of the three algorithms

Sample number	Evaluation result			True level (true value)
	LightGBM	GA-LightGBM	PRF-RFECV-GA-LightGBM	
134	Ultra-high risk (0.86)	Ultra-high risk (0.92)	Ultra-high risk (0.91)	Ultra-high risk (0.92)
175	Low risk (0.19)	Moderate risk (0.25)	Moderate risk (0.28)	Moderate risk (0.35)
178	Moderate risk (0.46)	Moderate risk (0.27)	Low risk (0.16)	Low risk (0.14)
284	Moderate risk (0.24)	Low risk (0.10)	Low risk (0.07)	Low risk (0.03)
315	Moderate risk (0.27)	Moderate risk (0.32)	High risk (0.53)	High risk (0.57)

(3) Comparison of algorithm evaluation metrics

As the comparison of the evaluation indexes of the three algorithms LightGBM, GA-LightGBM and PRF-RFECV-GA-LightGBM are shown in Table 3.

It can be seen that the root-mean-square error of PRF-RFECV-GA-LightGBM is reduced by 0.09 and 0.03 compared with LightGBM and GA-LightGBM, respectively, and the errors of PRF-RFECV-GA-LightGBM situational awareness model are lower than the other two models. Meanwhile, the R^2 of PRF-RFECV-GA-LightGBM is improved to 0.9232 by 0.0289 and 0.0071 than LightGBM and GA-LightGBM, respectively, and the evaluation metrics prove that the LightGBM algorithm optimized by PRF-RFECV and GA is trained better and with lower error.

Table 3: The three algorithm evaluation indicators

Model	MSE	RMSE	MAE	R^2
LightGBM	0.0033	0.0581	0.0168	0.8943
GA-LightGBM	0.0027	0.0541	0.0194	0.9161
PRF-RFECV-GA-LightGBM	0.0024	0.0525	0.0139	0.9232

(4) Comparison of the accuracy of situational level assessment

The evaluation accuracy rates of the three algorithms of PRF-RFECV-GA-LightGBM with GA-LightGBM and LightGBM for the posture level are shown in Figure 8. It can be seen that although GA-LightGBM has a good evaluation index of 94.55%, it is still slightly lower than the 97.96% of PRF-RFECV-GA-LightGBM after PRF-RFECV, and slightly higher than the unoptimized LightGBM algorithm in the accuracy rate of the corresponding posture level. The accuracy of the PRF-RFECV and GA-optimized LightGBM is 3.8 percentage points higher than that of the unoptimized LightGBM, which proves that the optimized model effectively improves the accuracy of the situational level assessment.

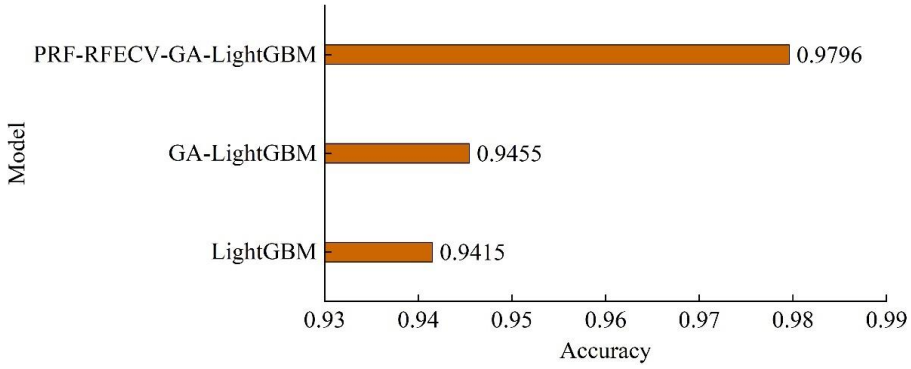

Figure 8: The accuracy rates of situation level assessment for the three optimization models

3 IPSO-BiLSTM based anomaly data prediction for power IoT network

On the basis of network security situational awareness, this chapter investigates anomalous data prediction for power IoT networks and proposes an anomalous data prediction model, IPSO-BiLSTM, which combines the Improved Particle Swarm Algorithm and BiLSTM.

3.1 Bidirectional Long Short-Term Memory Network (BiLSTM)

3.1.1 Long and Short Term Memory Networks (LSTM)

Long Short-Term Memory (LSTM) is a variant of Recurrent Neural Network (RNN), which introduces a gating mechanism to solve the gradient explosion or disappearance problem of traditional RNN in a simple and effective way, and the LSTM controls the information transfer between each cell through the gating mechanism. 3 “gates” are the forgetting gate, the input gate, and the output gate, denoted by and respectively. The three “gates” are forgetting gate, input gate and output gate, denoted by f_t, i_t and o_t respectively.

The internal structure of LSTM is shown in Figure 9. The internal structure of LSTM is shown in Fig. 9, where x_t denotes the input information at the current moment, h_{t-1} and h_t denote the output values of the cells at the previous and current moments, c_{t-1} and c_t denote the memory cells at the previous and current moments, σ represents the sigmoid activation function, and t represents the tangent function.

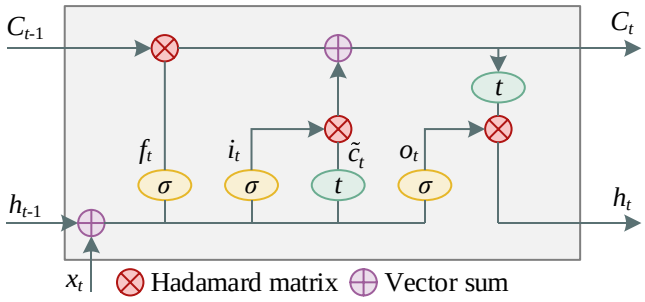

Figure 9: The basic structure of LSTM

Input gates are used to control the extent to which the cell needs to save information at the current moment:

$$\tilde{c}_t = \tanh\left(W_c \cdot \left[h_{t-1}, x_t\right] + b_c\right) \tag{33}$$

$$i_t = \sigma\left(W_i \cdot \left[h_{t-1}, x_t\right] + b_i\right) \tag{34}$$

The forgetting gate f_t is used to control the information discarded from the last moment memory cell c_{t-1}:

$$f_t = \sigma\left(W_f \cdot \left[h_{t-1}, x_t\right] + b_f\right) \tag{35}$$

With the input and output gates, the new cellular memory cell value at the current moment can be calculated:

$$c_t = f_t \otimes c_{t-1} + i_t \otimes \tilde{c}_t \tag{36}$$

The output gate is used to control the amount of information that the cell memory cell value at the current moment is used as the output value for that cell:

$$o_t = \sigma\left(W_o \cdot \left[h_{t-1}, x_t\right] + b_o\right) \tag{37}$$

$$h_t = o_t \otimes \tanh\left(c_t\right) \tag{38}$$

In Eqs. (33)~(38), W and b represent the weight matrix and bias term, respectively, σ represents the sigmoid activation function, tanh represents the tangent function, and $\otimes$ represents the matrix Hadamard product.

3.1.2 Bidirectional Long Short-Term Memory Network (BiLSTM)

LSTM is unidirectional to extract sequence information, but for the problem of anomalous data prediction in power IoT networks, the data status of the network at the current moment is not only related to the status at the previous moment, but also may be related to the status afterward. In order to improve the prediction effect, this paper introduces the bidirectional long and short-term memory network (BiLSTM).

The output of BiLSTM is jointly determined by two LSTM layers, the forward LSTM layer can be regarded as the forward computation from the starting moment to the last moment, and the reverse LSTM layer can be regarded as the reverse computation from the last moment to the starting moment, which are processed in the same way in the two layers. Finally, the outputs of the forward and reverse layers at each moment are combined to obtain the output at that moment. For BiLSTM networks, the choice of parameters in its structure is crucial to the effect of the model. Therefore, in this paper, the particle swarm algorithm, which is simple in principle, low in complexity, fast in convergence, and suitable for dealing with real-valued problems, is selected to optimize the structural parameters of the BiLSTM network.

3.2 Improved particle swarm algorithm

3.2.1 Particle Swarm Optimization

Particle swarm algorithm is a bionic swarm optimization algorithm, which originated from the

study of regular feeding behavior of bird flocks. The basic idea of the particle swarm algorithm is to treat each solution of the problem as a D-dimensional massless particle, and each particle carries a fitness value determined by a fitness function. In the search space, each particle updates its own speed and position according to the individual optimal position and the global optimal position, and the optimal position of the whole particle swarm is obtained through iterative search.

In each iteration, the particles in the swarm determine their search direction and distance by their velocity, and the updating formula for the velocity and position of the elementary particle swarm is as follows:

$$V_{id}^{k+1} = wV_{id}^{k} + c_1 r_1 \left(pbest_{id}^{k} - X_{id}^{k} \right) + c_2 r_2 \left(gbest_{gd}^{k} - X_{id}^{k} \right) \tag{39}$$

$$X_{id}^{k+1} = X_{id}^{k} + V_{id}^{k+1} \tag{40}$$

where k represents the current number of iterations. w represents the inertia weight factor, which is the ability of the particle to inherit the speed of the previous iteration. c_1 and c_2 represent the acceleration factors, which are used to regulate the effect of the individual optimal solution and the global optimal solution of each iteration on the velocity. r_1 and r_2 are random numbers between $[0,1]$. V_{id}^{k} and X_{id}^{k} represent the velocity and position of the i th particle in the d th dimension at the k th iteration, respectively. $pbest_{id}^{k}$ and $gbest_{gd}^{k}$ represent the individual optimal position and global optimal position of the d-dimensional space of the i th particle at the k th iteration, respectively.

3.2.2 Particle Swarm Algorithm Improvement

In particle swarm algorithms, the role of inertia weighting factors and acceleration factors is crucial to the efficiency and results of PSO algorithms. Since the inertia weighting factor and acceleration factor in the traditional particle swarm algorithm are fixed, it limits the local and global optimization ability of the algorithm, and it is also easy to make the algorithm fall into the local minimum. Aiming at the limitations of the algorithm, this paper improves the inertia weighting factor and acceleration factor, so that the change of the speed is changed from linear to nonlinear.

The improvement of the inertia weight factor w is as follows:

$$w = -\pi * arc\ \sin\left(0.01*\left(t - \max\_iter\right)\right) \tag{41}$$

Improvements to the values of the acceleration factors c_1 and c_2 are as follows:

$$c_1 = c_{1\max} - (c_{1\max} - c_{1\min}) * (t / \max\_iter) 2 \tag{42}$$

$$c_2 = c_{2\max} - (c_{2\max} - c_{2\min}) * (t / \max\_iter) 2 \tag{43}$$

where t represents the current number of iterations, $\max\_iter$ represents the maximum number of iterations, $c_{1\max}$ and $c_{1\min}$ represent the maximum and minimum values of c_1 respectively, and $c_{2\max}$ and $c_{2\min}$ represent the maximum and minimum values of c_2, respectively.

3.3 Construction of IPSO-BiLSTM prediction model

3.3.1 Construction of the BiLSTM model

The structure of the BiLSTM model used in the experiments of this paper is shown below:

(1) BiLSTM layer: with two BiLSTM layers, its ability to combine before and after can be fully utilized to enhance the model learning.

(2) Dropout layer: avoid model overfitting and improve generalization ability.

(3) Dense layer: set the last layer as Dense to convert the output dimension and get the prediction result.

3.3.2 IPSO-BiLSTM Network Anomaly Data Prediction Models

The IPSO-BiLSTM model optimizes the relevant parameters of BiLSTM according to the fast optimization-seeking ability of the IPSO algorithm to improve the anomaly data prediction effect of BiLSTM.

The flow of the IPSO-BiLSTM power IoT network anomaly data prediction model is shown in Fig. 10, and its specific steps are as follows:

Step 1: Construct the training set samples and test set samples according to the size of the sliding window.

Step 2: Initialize the relevant parameters in IPSO, including the search dimension D, the number of particles pN, the maximum and minimum values of the acceleration factors c_1 and c_2, the maximum number of iterations $\max\_iter$, the initial position of the particles X_i^0 and the initial velocity V_i^0, the inertia weighting factor w and the learning factors r_1 and r_2 are automatically generated in the iteration.

Step 3: Set the range of values of each dimension in the particle to be optimized, and the particle dimensions $\alpha, iterator, n_1, n_2$ and s represent the learning rate, the number of iterations of the model, the number of units of the first implicit layer, the number of units of the second implicit layer of the LSTM, and the random seed in the BiLSTM model, respectively.

Step 4: Set the fitness function of the particle swarm algorithm, randomly generate the initial position of the particle swarm, calculate the initial fitness value of each particle, and obtain the individual optimal solution $pbest_{id}^k$ and the global optimal solution $gbest_{gd}^k$ at the initial time.

Step 5: Calculate the fitness value of each particle, update the individual optimal solution $pbest_{id}^k$ and global optimal solution $gbest_{gd}^k$, and calculate the velocity of the particle and update the position of the particle according to Eqs. (41)~(43).

Step 6: If the maximum number of iterations is reached, proceed to step 7, otherwise return to step 5 to continue iteration.

Step 7: The obtained optimal parameters are given to the BiLSTM model to obtain the anomaly data prediction results.

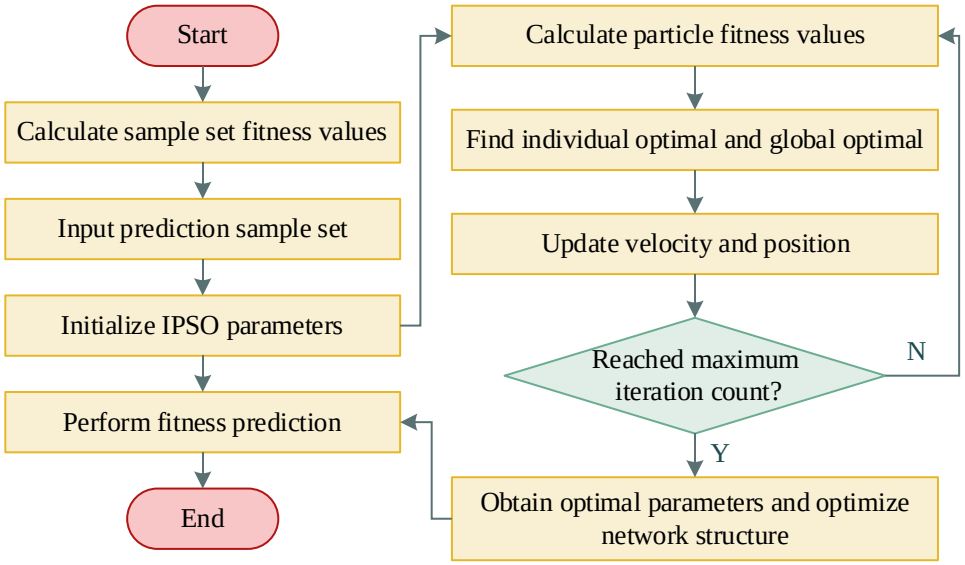

Figure 10: IPSO-BiLSTM prediction process

3.4 Experimental analysis

3.4.1 Experimental environment

Experimental computer configuration: processor Intel(R) Core(TM) i7-13700H, graphics card RTX4060Ti, hard disk 1TB, memory 64GB. Environment for Window11 operating system, 64-bit, simulation software using Pycharm, using Python3.13 to build the Keras framework for experiments.

3.4.2 Evaluation indicators

In this experiment, MAE, MAPE, RMSE and other indicators are used as indicators to evaluate the prediction performance of the model, and the smaller the value of the indicator indicates the higher prediction accuracy.

3.4.3 Analysis of experimental results

The comparison of the prediction model IPSO-BiLSTM with the true value, BiLSTM, IPSO-SVM, and PSO-BiLSTM used in this paper is shown in Figs. 11~13. Among them, the anomalous data classes A~E indicate fatal, severe, general, slight, and potential, respectively. It can be found that the accuracy rate of the IPSO-BiLSTM model is significantly higher than that of the BiLSTM and PSO-BiLSTM models, which is almost the same as that of the IPSO-SVM model.

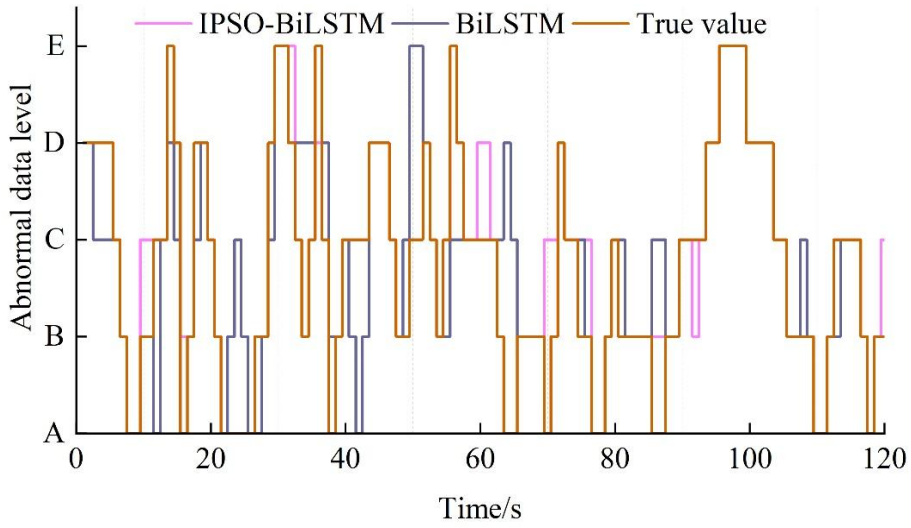

Figure 11: Comparison of IPSO-BiLSTM with BiLSTM and true values

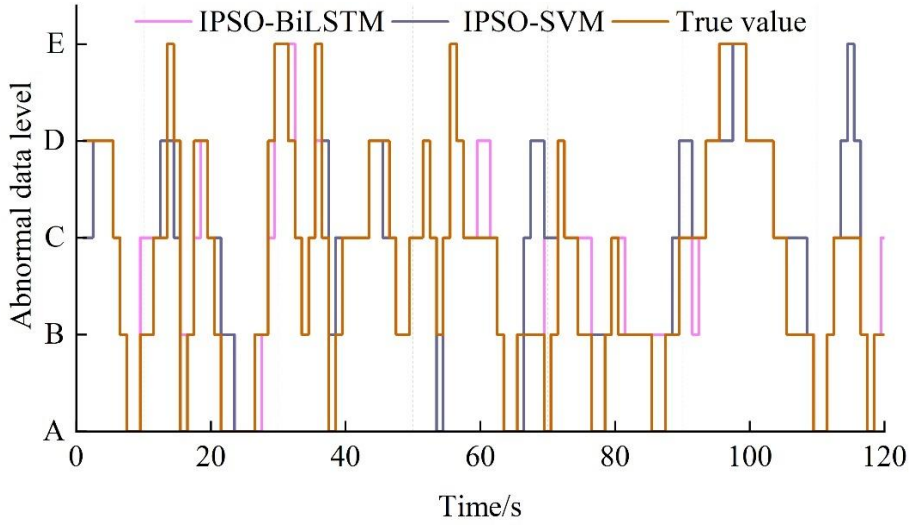

Figure 12: Comparison of IPSO-BiLSTM with IPSO-SVM and true values

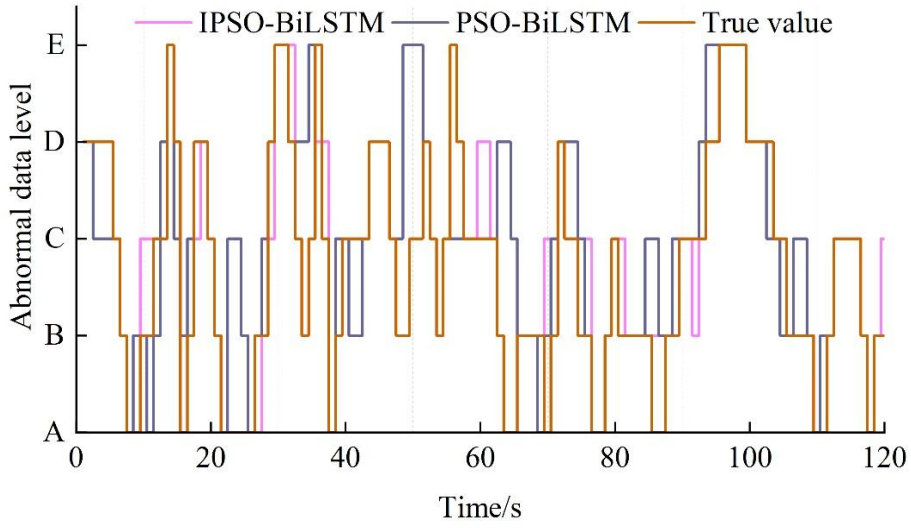

Figure 13: Comparison of IPSO-BiLSTM with PSO-BiLSTM and true values

By comparing the difference of the prediction models IPSO-BiLSTM, BiLSTM, IPSO-SVM, PSO-BiLSTM used in this paper with the real values respectively and then one by one, the comparison of the difference of IPSO-BiLSTM model with the difference of the three models of BiLSTM, IPSO-SVM, and PSO-BiLSTM are shown in Figs. 14~16 respectively.

It can be seen that the difference of IPSO-BiLSTM is more stable than that of BiLSTM, which means that the error between the predicted value and the real value of IPSO-BiLSTM is smaller.The degree of stability of the difference of IPSO-BiLSTM is basically the same as that of IPSO-SVM, but the error points of IPSO-SVM are obviously more than that of IPSO-BiLSTM That is to say, although the stability of IPSO-BiLSTM is almost the same as that of IPSO-SVM, the accuracy is significantly higher than that of IPSO-SVM, and the difference of IPSO-BiLSTM is more stable than that of PSO-BiLSTM, and the error points of IPSO-BiLSTM are significantly less than those of PSO-BiLSTM, which indicates that the performance of IPSO-BiLSTM model is better than that of IPSO-SVM. BiLSTM model performs better than PSO-BiLSTM, and the improvement method of particle swarm algorithm designed in this paper is effective.

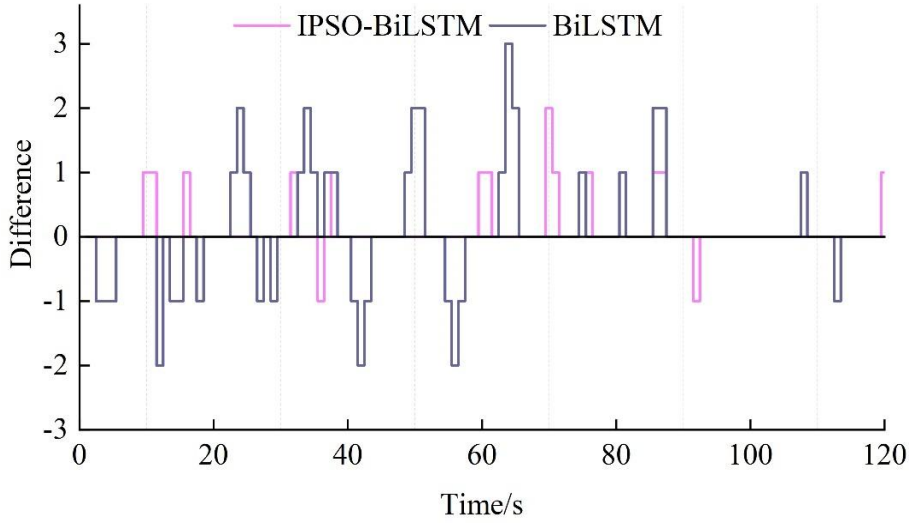

Figure 14: Comparison of IPSO-BiLSTM difference and BiLSTM difference

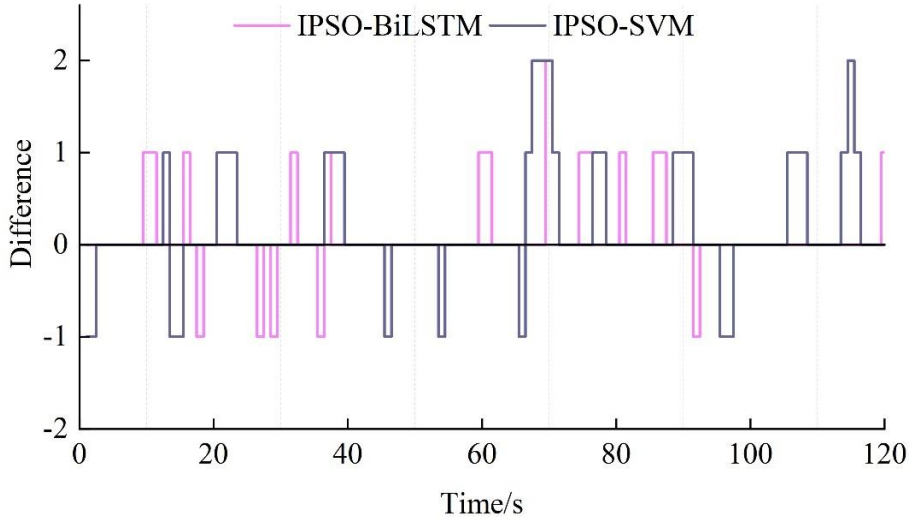

Figure 15: Comparison of IPSO-BiLSTM difference and IPSO-SVM difference

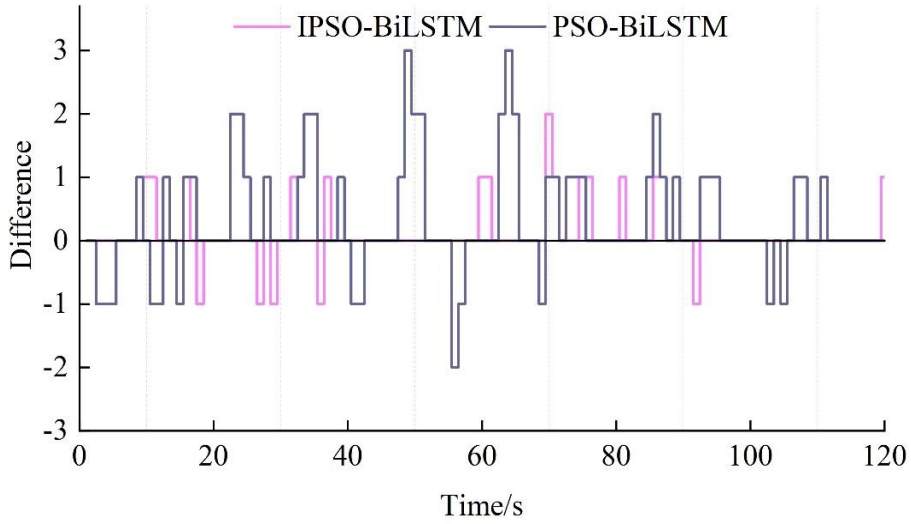

Figure 16: Comparison of IPSO-BiLSTM difference and PSO-BiLSTM difference

After calculation, the metrics of prediction accuracy of several models are shown in Table 4.

Comparing the MAE, MAPE, and RMSE metrics of the four models, IPSO-BiLSTM, BiLSTM, IPSO-SVM, and PSO-LSTM, respectively, it is obtained that in this metric of MAE IPSO-BiLSTM is 0.84, 0.96, and 1.00 lower than the other three models. In the metric of MAPE, the other three models outperform the IPSO-BiLSTM model by 0.05, 0.11, and 0.04, respectively.In the metric of RMSE, IPSO-BiLSTM outperforms the other three models by 3.37, 6.40, and 5.48.

Table 4: Results of prediction performance indicators of different models

Model	MAE	MAPE	RMSE
IPSO-BiLSTM	3.28	0.12	8.87
IPSO-SVM	4.12	0.17	12.24
BiLSTM	4.24	0.23	15.27
PSO-BiLSTM	4.28	0.16	14.35

Comprehensively, the above charts show that IPSO-SVM is a support vector machine model optimized by the Improved Particle Swarm Algorithm, which uses the Improved Particle Swarm Algorithm to optimize the penalty function and the kernel function of the SVM to improve the accuracy of the model.As a traditional method of machine learning, the main performance of the model lies in the use of the kernel function and the final decision depends on the small number of support vectors in a large probability. For complex network environments and large amounts of data, the performance of SVM decreases. The use of SVM as the main model is time-consuming and computationally expensive when solving cyber security related problems.

The original BiLSTM model is simpler and more affected by the parameters, and setting different parameters in different environments has a serious impact on the results. The BiLSTM model alone has low accuracy and low training efficiency, which obviously does not meet the experimental requirements.The PSO-BiLSTM model is an optimization of BiLSTM using the particle swarm algorithm.

The IPSO algorithm balances the global search ability and local search ability by improving the inertia weight factor and acceleration factor, which can find the optimal solution faster, and the effect is better than the PSO algorithm.PSO-BiLSTM model and

IPSO-BiLSTM model belong to the improved model of swarm intelligent optimization algorithm, and we can see that there is a big difference in accuracy between the two optimization models as far as the accuracy is concerned, and the error rate of IPSO-BiLSTM model is lower than that of IPSO-BiLSTM model as far as the error rate is concerned. IPSO-BiLSTM model is slightly better in terms of error rate. Therefore, the overall performance of the IPSO-BiLSTM model used in this paper is better, which basically meets the experimental needs and can accurately predict the abnormal data of electric power IOT network.

4 Conclusion

In this paper, based on the machine learning network security situational awareness model PRF-RFECV-GA-LightGBM, we construct a network anomaly data prediction model based on IPSO-BiLSTM, and evaluate the performance of the two models separately.

Compared with the three algorithms of SVM, DNN, and XGBoost, LightGBM is more similar to the real value and real grade both in terms of the evaluated posture value and the corresponding posture grade. However, the LightGBM algorithm has the lowest error, and the R^2 value of 0.8943 and the accuracy rate of 0.9415 are the highest among the four algorithms, so the selection of LightGBM as the base algorithm in this paper is reasonable. Meanwhile, the situational awareness results of the constructed PRF-RFECV-GA-LightGBM model are basically the same as the real grade, which is better than LightGBM and GA-LightGBM. and the PRF-RFECV-GA-LightGBM model has a lower error, and the coefficient of determination, R^2, and the accuracy rate, 0.9232 and 0.9796, respectively, are more achieved with the better values, indicating that the model is suitable for power logistics network security situational awareness tasks.

Meanwhile, compared with BiLSTM, IPSO-SVM, PSO-LSTM and other models, the IPSO-BiLSTM model has the lowest MAE, MAPE and RMSE, and the prediction results of the network anomaly data are closer to the real values, and the prediction is more stable, which is more suitable for the increasingly complex network environment.

The algorithm proposed in this project has been experimentally proved to have made good progress in the accuracy of situational awareness and anomaly data prediction, but there are still some shortcomings:

(1) The proposed algorithm has selected the existing dataset to validate the model, and subsequently needs to continue to be tested in a real power IOT network environment to further improve its performance.

(2) The rank accuracy of situational awareness and anomalous data prediction still needs to be improved, and the subsequent search for a more optimized way to continue to improve the accuracy of the prediction of anomalous data, in order to achieve better situational awareness.

Funding

This work was supported by ”Research and Application of Key Technologies for Smart Power Supply and Guarantee Digital Platform”,Guizhou Power Grid Co., Ltd. Electric Power Research Institute, Guizhou, China, grant number 060000KC23100042

About the Author

Guobang Ban (1982-11), male, Bouyei ethnicity, native of Zhijin, Guizhou Province, holds a master's degree. North China Electric Power University, Senior Engineer, specializing in operational risk prevention and control as well as emergency equipment technology for power grid enterprises.

Yumin He (1995-9), male, Han ethnicity, from Guiyang, Guizhou Province, holds a master's degree. East China University of Science, Engineer, specializing in power grid automation.

Guanghui Xi (1985-9), male, Han ethnicity, from Funan, Anhui Province, holds a master's degree. North China Electric Power University, Senior Engineer, the research direction is emergency power supply and safety production of power grid enterprises.

Xinbiao Xiong (1998-12), male, Tujia ethnicity, native of Tongren, Guizhou Province, holds a master's degree from Guizhou University. Assistant Engineer, specializing in emergency power supply and production safety for power grid enterprises.

Jiangang Liu (1986-2), male, Han ethnicity, from Anshun, Guizhou Province, holds a bachelor's degree. Mingde College, Guizhou University, Engineer, specializing in operational risk management for power grid enterprises, intelligent management technologies for accident incidents, and occupational safety.

Siqi Guo (1996-10), female, Tujia ethnicity, from Tongren, Guizhou Province, holds a bachelor's degree and works as an engineer. Her research focuses on the application of ubiquitous safety management in the power industry and related fields.

References

[1] Jiang, A., Yuan, H., Li, D., & Tian, J. (2019). Key technologies of ubiquitous power Internet of Things-aided smart grid. Journal of Renewable and Sustainable Energy, 11(6).

[2] Yaqoob, I., Ahmed, E., Hashem, I. A. T., Ahmed, A. I. A., Gani, A., Imran, M., & Guizani, M. (2017). Internet of things architecture: Recent advances, taxonomy, requirements, and open challenges. IEEE wireless communications, 24(3), 10-16.

[3] Kumar, N. M., & Mallick, P. K. (2018). The Internet of Things: Insights into the building blocks, component interactions, and architecture layers. Procedia computer science, 132, 109-117.

[4] Færøy, F. L., Yamin, M. M., Shukla, A., & Katt, B. (2023). Automatic verification and execution of cyber attack on IoT devices. Sensors, 23(2), 733.

[5] Msgna, M. (2022). Anatomy of attacks on IoT systems: review of attacks, impacts and countermeasures. Journal of Surveillance, Security and Safety, 3(4), 150-173.

[6] Savukynas, R. (2020, June). Internet of Things information system security for smart devices identification and authentication. In 2020 9th Mediterranean Conference on Embedded Computing (MECO) (pp. 1-5). IEEE.

[7] Rozlomii, I., Yarmilko, A., & Naumenko, S. (2024). Data security of IoT devices with limited resources: challenges and potential solutions. doors, 3666, 85-96.

[8] Yuan, B., Yang, M., Xu, Z., Chen, Q., Song, Z., Li, Z., ... & Jin, H. (2023). Leakage of authorization-data in IoT device sharing: New attacks and countermeasure. IEEE Transactions on Dependable and Secure Computing, 21(4), 3196-3210.

[9] Ahmed, M., & Choudhury, S. (2018). False data injection attacks in internet of things. In Performability in internet of things (pp. 47-58). Cham: Springer International Publishing.

[10] Kolisnyk, M. (2021). Vulnerability analysis and method of selection of communication protocols for information transfer in Internet of Things Bostami, B.,systems. Radioelectronic and computer systems, (1), 133-149.

[11] Sun, L., Zhou, K., Zhang, X., & Yang, S. (2018). Outlier data treatment methods toward smart grid applications. IEEE Access, 6, 39849-39859.

[12] Fahim, M., & Sillitti, A. (2019). Anomaly detection, analysis and prediction techniques in iot environment: A systematic literature review. IEEE Access, 7, 81664-81681.

[13] Gaddam, A., Wilkin, T., Angelova, M., & Gaddam, J. (2020). Detecting sensor faults, anomalies and outliers in the internet of things: A survey on the challenges and solutions. Electronics, 9(3), 511.

[14] Cheng, M., Zhang, D., Yan, W., He, L., Zhang, R., & Xu, M. (2023). Power system abnormal pattern detection for new energy big data. International Journal of Emerging Electric Power Systems, 24(1), 91-102.

[15] Ahmed, S., Lee, Y., Hyun, S. H., & Koo, I. (2019). Unsupervised machine learning-based detection of covert data integrity assault in smart grid networks utilizing isolation forest. IEEE Transactions on Information Forensics and Security, 14(10), 2765-2777.

[16] Kabir, S., Hannan, N., Shufian, A., & Zishan, M. S. R. (2025). Proactive detection of cyber-physical grid attacks: A pre-attack phase identification and analysis using anomaly-based machine learning models. Array, 100441.

[17] Choi, J., Roshanzadeh, B., Martínez-Ramón, M., & Bidram, A. (2023). An unsupervised cyberattack detection scheme for AC microgrids using Gaussian process regression and one-class support vector machine anomaly detection. IET Renewable Power Generation, 17(8), 2113-2123.

[18] Aboelwafa, M. M., Seddik, K. G., Eldefrawy, M. H., Gadallah, Y., & Gidlund, M. (2020). A machine-learning-based technique for false data injection attacks detection in industrial IoT. IEEE Internet of Things Journal, 7(9), 8462-8471.

[19] Atassi, R. (2023). Anomaly Detection in IoT Networks: Machine Learning Approaches for Intrusion Detection. Fusion: Practice & Applications, 13(1).

[20] Miao, W., Zhao, X., Zhang, Y., Chen, S., Li, X., & Li, Q. (2024). A Deep Learning-Based method for preventing data leakage in electric power industrial internet of things business data interactions. Sensors, 24(13), 4069.

[21] Urs, P. M., Reddy, A. T. N., Mallikarjunaswamy, S., & Lakshminarayan, U. M. (2025). An Innovative IoT Framework using Machine Learning for Predicting Information Loss at the Data Link Layer in Smart Networks. Engineering, Technology & Applied Science Research, 15(2), 20904-20911.

[22] Malki, A., Atlam, E. S., & Gad, I. (2022). Machine learning approach of detecting anomalies and forecasting time-series of IoT devices. Alexandria Engineering Journal, 61(11), 8973-8986.

[23] Zheng, S., Cheng, J., Xiong, H., Wang, Y., & Wang, Y. (2024). Big Data Anomaly Prediction Algorithm of Smart City Power Internet of Things Based on Parallel Random Forest. Journal of Testing and Evaluation, 52(3), 1429-1442.

[24] Hao, W., Yang, Q., Li, Z., Hu, S., Liu, B., & Ruan, W. (2022). Multi-scale traffic aware cybersecurity situational awareness online model for intelligent power substation communication network. IEEE Internet of Things Journal, 10(2), 1666-1681.

[25] Ji, Y., Jin, M., Hu, Y., Liu, X., & Jin, Q. (2024, March). Research on Data Interaction Behavior Security Situational Awareness of Power monitoring system. In 2024 IEEE 7th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC) (Vol. 7, pp. 653-658). IEEE.

[26] Almeida, R. B., Junes, V. R. C., da Silva Machado, R., da Rosa, D. Y. L., Donato, L. M., Yamin, A. C., & Pernas, A. M. (2019). A distributed event-driven architectural model based on situational awareness applied on internet of things. Information and software technology, 111, 144-158.

[27] Wang, C., Dong, J. H., Guo, G. X., Ren, T. Y., Wang, X. H., & Pan, M. Y. (2023). Security situational awareness of power information networks based on machine learning algorithms. Connection Science, 35(1), 2284649.

[28] Zhen, L., Kamarudin, N. H., Kok, V. J., & Qamar, F. (2025). Anomaly detection model in network security situational awareness based on machine learning: Limitation, techniques, future trends. IEEE Access.