Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105
Volume **11**, Number **1**, February **2026**

Intrusion Risk Detection Model for Power Systems Based on Deep Neural Networks and Integrated Learning

Liang Meng[1,*]

[1]Information Communication Branch of Guangxi Grid Company,
Nanning 530012, P. R. China
452899946@qq.com

Xin Liu[2]

[2]CSG Digital Power Grid Group Co., Ltd.,
Guangzhou 510000, P. R. China
chen1998mail@163.com

Ming Xie[3], Li-Na Chen[3] and Jun-Hao Song[3]

[3]Information Communication Branch of Guangxi Grid Company,
Nanning 530012, P. R. China
chenhuizuiweida@qq.com, 403367210@qq.com, chen1998mail@foxmail.com

[*]Corresponding author: Liang Meng

Received October 12, 2024, revised January 27, 2025, accepted April 11, 2025.

Abstract. *The traffic size of power system is growing rapidly, and the direct and indirect threats of network attacks on power grid security are becoming more and more prominent. Existing intrusion risk detection methods for power systems have exposed the defects of difficult data feature extraction, poor generalization ability, low detection rate, and so on. To this end, this paper designs a power system intrusion risk detection model relied on deep neural network and integrated learning. Firstly, the combination strategy of subtrees in the Random Forest (RF) integrated learning algorithm is improved, and the decision weight of the base classifier is re-quantified based on the exponentially weighted calculation, so that the skewed influence of the trees with lower accuracy rates is reduced before final detection. Then the local features of power system network traffic are extracted using CNN, and the localized features are fed into the BiLSTM network for global learning, which can extract global features containing long-term temporal attributes, and the attention mechanism is introduced to highlight important features. Finally, the cascaded RF integration learning algorithm is utilized to integrate the predictions of each base classifier to output the final detection results. Experimental outcome on the CIC-IDS-2017 dataset show that the detection accuracy of the designed model improves by* $2.89\% - 24.66\%$ *compared to the comparison model, and it can efficiently accomplish the task of power system intrusion risk detection.*

Keywords: Intrusion risk detection; Deep neural network; Random Forest; Convolutional neural network; BiLSTM.

1. **Introduction.** Power system plays an essential role in social life, as the information and communication technologies rapidly growing, China has entered the era of digital power, and the power information network, as the infrastructure of the power system, offers immense convenience to the work of the enterprise [1]. Along with the development

of smart grid, various network attacks and illegal invasion phenomena are becoming more and more common, and the means of invasion are becoming more and more complex. Network intrusion behavior will lead to power system data leakage, which seriously affects the operation of the power system, how to stop the abnormal traffic invasion, to protect the security of the power information system has been widely concerned by many scholars [2, 3]. Intrusion risk detection is the basic method used to identify different network attacks in the system, and it is very important to improve the efficiency of intrusion risk detection. Intrusion detection for power system networks enables real-time defense against internal, external and erroneous operations before the network is attacked, and effectively intercepts and defends against them [4].

1.1. **Related work.** Traditionally, intrusion detection risk modeling is performed in power systems by manually constructing rules. Pan et al. [5] who used sequential pattern mining algorithms for detecting disturbances and cyber-attack behaviors in power systems. Umar and Felemban [6] proposed a rule model-based intrusion detection technique for power systems that detects changes in the availability of servers and services themselves by specifying expected characteristics of network requests and responses based on the Modbus protocol. This type of research relies heavily on manually formulated rules, and machine-learning based methods can automatically generate rules from existing data without human intervention. Upadhyay et al. [7] used Support Vector Machine (SVM) algorithm to detect anomalous activities in power system traffic, but the accuracy of detection is low. Rose et al. [8] used K-means to classify the internal anomalies and finally used one-class SVM for detection. Azam et al. [9] first selects the power system network traffic feature data and in order to achieve data dimensionality reduction, and then uses a decision tree model to classify the attacks, the classification accuracy is only 78.3%.

However, while ML algorithms are extremely dependent on a professional to extract features, deep neural networks can take valid features directly from raw data and make them high-level. Radoglou et al. [10] combined Artificial Neural Networks (ANN) with a simple frequency-based coding scheme for detecting potential power system intrusions. Song et al. [11] used LSTM network for feature extraction and embedded Decision Tree (DT) model in SVM for anomalous traffic detection. Convolutional neural network (CNN) has also gained wider use in intrusion risk monitoring in power systems due to their local sensing and weight sharing, which can greatly reduce the number of parameters. Gao et al. [12] first used different dimensionality reduction methods to remove redundant features, and then passed the data obtained after dimensionality reduction to CNN, but ignored the advantage of CNN to extract features automatically. Zhai et al. [13] proposed an intrusion detection model based on the combination of inflated convolution and gated recurrent unit (GRU) and simulated it in NSL-KDD dataset and achieved a high accuracy rate. Zhang et al. [14] utilized CNN to learn the underlying spatial-temporal characteristics of the power system traffic and output the intrusion risk detection results via sigmoid, the detection accuracy rate reached 86.9%.

Although deep neural networks have strong feature extraction capabilities, the complexity of the classifiers used is high, integrated learning aims to combine multiple classifiers in a certain way, which greatly improves the efficiency of intrusion risk detection, representative algorithms are Random Forest (RF) [15], AdaBoost [16], XGBoost [17] and so on. Zhu et al. [18] designed an integrated studying model for area under ROC curve random forest algorithm to improve the effect of intrusion detection. Panthi et al. [19] used particle swarm algorithm PSO to optimize the hyperparameters of XGBoost algorithm, and then used the combined model for intrusion detection, and proved the validity of the model through experiments.

1.2. **Contribution.** Aiming at the above issues of feature extraction difficulty, poor generalization ability and low detection correctness of the power system intrusion detection model, this paper designs a power system intrusion risk detection model based on deep neural network and integrated learning. Firstly, re-quantifying the weight of base classifiers' decisions in RF based on the calculation of exponential weights to reduce the skewed influence of base classifiers with lower prediction accuracies on the prediction results of the whole integrated learning. Then CNN is adopted to capture local characteristics of power system network traffic, the local features are fed into BiLSTM network for global learning, global features containing temporal attributes are extracted by capturing the backward and forward sequential dependencies among the features, and an attention mechanism is introduced to reevaluate the input features so as to give a new weight to the traffic features in order to ignore irrelevant features. Finally, the detection results are output through the optimized RF integration learning method. The experimental outcome indicates that the designed model has high detection accuracy, check accuracy and check completeness, and has good detection effect.

2. Theoretical analysis.

2.1. **Intrusion risk detection theory.** Intrusion risk detection monitors the real-time traffic in the network [20], determines whether there is anomalous behavior into the system, and if so, alarms and prevents the entry of anomalous behavior, as shown in Figure 1, and the main roles of RF are as follows:

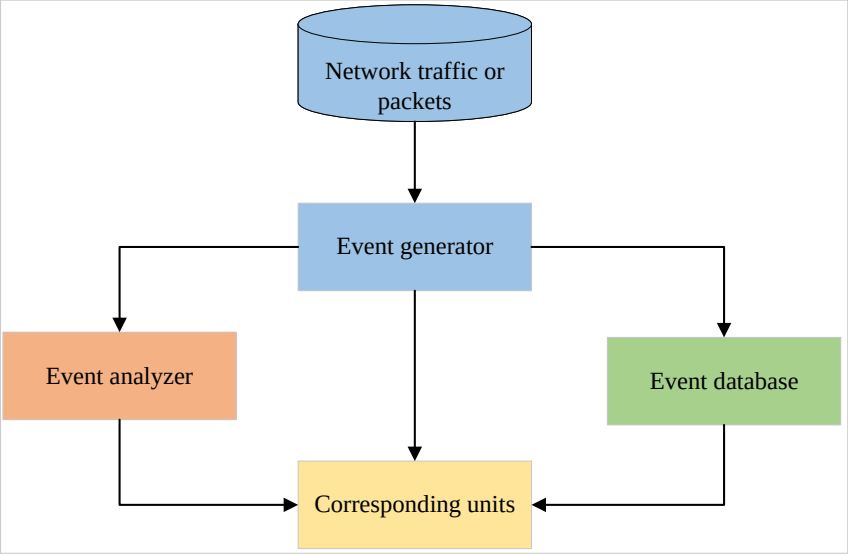

Figure 1. Intrusion risk detection system working mode

(1) Detect abnormal and malicious behaviors in the network and detect intrusions or anomalies in a timely manner through techniques such as network traffic analysis, log analysis and malicious behavior analysis.
(2) By analyzing network traffic, we can detect invasive behavior or anomalies, such as virus attacks and malicious programs, and take timely defensive measures.
(3) When intrusion behavior or anomalies are detected, the system will record the relevant information and issue timely warnings to prevent data leakage and system paralysis.

2.2. **Convolutional neural network.** CNN is a deep neural network, compared with the traditional ML algorithm structure, CNN can automate the learning and extraction of features, so it has the advantages of high classification accuracy and computational efficiency. The core structure of CNN is made up of alternating convolutional and pooling levels [21], and the basic structure is that after multiple convolutional and pooling operations the data is output through a fully connected level. The convolutional level realizes the feature extraction of the dataset through convolution operation, which is computed as below:

$$f(x) = \sum_{ij}^{n} \theta_{ij} \times x_{ij} + b \tag{1}$$

where $f(x)$ is the output eigenvalue, θ_{ij} is the value of the element in row i and column j of the filter, x_{ij} is the value of the corresponding element of the input data, and b is the bias.

The pooling level is an operation that further compresses the values input from the convolutional level, such as Max pooling and average pooling.

The fully connected level is to classify the features, in this level the features are expanded in the form of vectors, which are output after nonlinear combinatorial operations, which are given in the following formulas:

$$f(x) = W \times x + b \tag{2}$$

where x is the output of the pooling level, W is the weight, b is the bias.

2.3. **Random forest.** RF is a classical integrated learning algorithm whose advantages mainly include high accuracy, robustness to noise and outliers, not prone to overfitting, and fast training speed. RF uses self-service resampling to create multiple subsets of the original training set to build a "forest" [22], which can be formalized as $D = \{(x_i, y_i), i = 1, 2, \ldots, N, \text{ and } j = 1, 2, \ldots, M\}$. The main process of building the random forest algorithm is shown in Figure 2.

(1) Generate k training subsets from D by sampling with Bootstrap sampling. Each sampling randomly draws N data from D with putback. k training subsets form the set $D_{Train} = \{D_1, D_2, \ldots, D_k\}$. At the same time, the unselected data in each sampling cycle are formed into a dataset named out-of-bag (OOB), k OOB datasets constitute the set $S_{OOB} = \{OOB_1, OOB_2, \ldots, OOB_k\}$.
(2) Create a DT on each D_i. Start splitting from the root node and keep repeating this computational process until leaf nodes are generated.
(3) Combine k trained DTs into a random forest as shown in Equation (3). x is the feature vector of the training data, and Θj is an independent and identically distributed random vector. $H(X, \Theta j)$ is the final prediction result.

$$H(X, \Theta j) = \sum_{i=1}^{k} h_i(x, \Theta j) \tag{3}$$

3. **Optimization of random forest integrated learning algorithm based on exponential weights.** It has been demonstrated that RF is one of the best classification algorithms in the field of intrusion detection [23]. However, the prediction accuracy of each DT in RF is different, and classical RF algorithms use simple arithmetic averaging and voting methods to integrate subtrees, which can skew the predictions when integrating

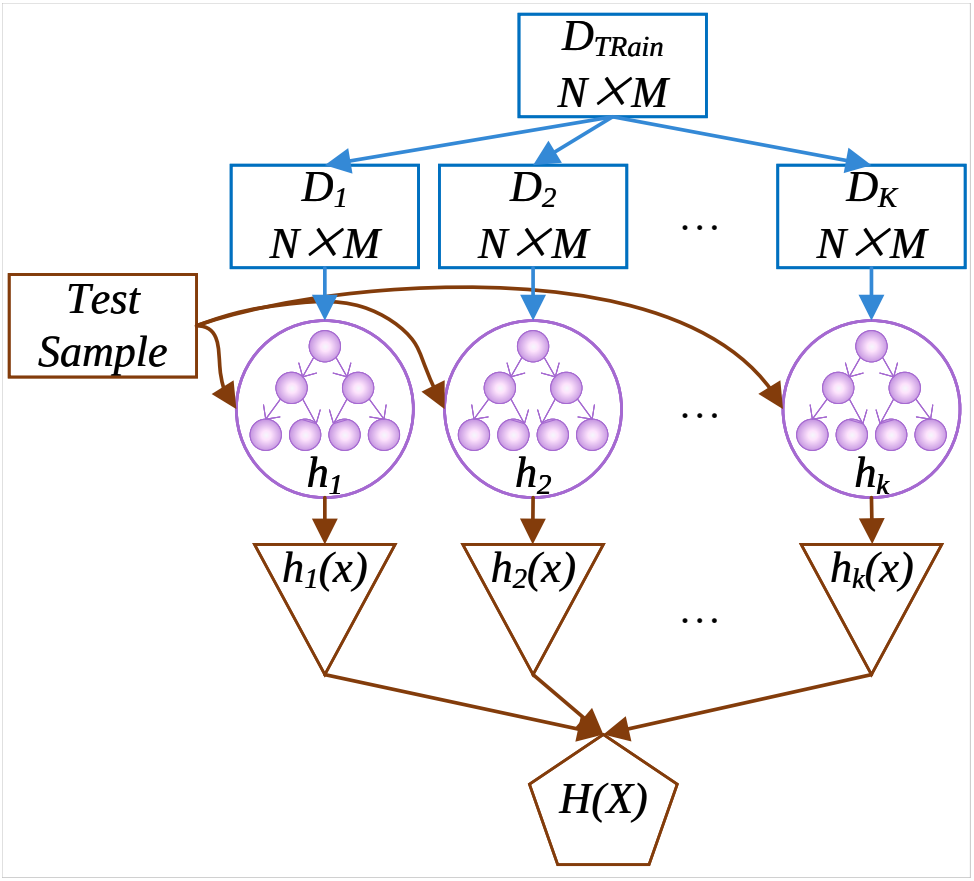

Figure 2. The main process of the RF algorithm

base classifiers with lower prediction accuracies. Thus, this paper optimizes the combining strategy for DT integration and propose an RF algorithm based on exponentially weighted optimization (WORF).

Let the number of tuples contained in the training set D and test set S be m and n, respectively, and the amount of labeling categories to which D and S belong is defined as c. The set of category labels is denoted as $L = \{l_1, l_2, l_3, \ldots, l_c\}$. Define the RF containing t decision trees, the k-th subtree in the RF is denoted as $T_k(k \in [1, t])$, i and j are count variables. Then the probability that the i-th training sample in D is predicted by T_k to be in category l_j is p_{ij}. The probability matrix of T_k's prediction of D is as follows:

$$\text{Probability}(T_k, D) = \begin{pmatrix} p_{11} & p_{12} & p_{13} & \cdots & p_{1c} \\ p_{21} & p_{22} & p_{23} & \cdots & p_{2c} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{m1} & p_{m2} & p_{m3} & \cdots & p_{mc} \end{pmatrix} \tag{4}$$

Define the function $RowMax(X)$ as obtaining the column subscript values corresponding to the maximum value of the row vector X of a two-dimensional matrix. Then the prediction of T_k on D is as follows:

$$\text{Predict}(T_k, D) = \begin{pmatrix} RowMax(p_{11} & p_{12} & \ldots & p_{1c}) \\ RowMax(p_{21} & p_{22} & \ldots & p_{2c}) \\ \vdots & \vdots & \ddots & \vdots \\ RowMax(p_{m1} & p_{m2} & \ldots & p_{mc}) \end{pmatrix} = \begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_m \end{pmatrix} \tag{5}$$

Define the function $Acc(X_1, X_2)$ to calculate the ratio of the number of equal elements in the same position in two vectors X_1 and X_2 of the same dimension to all the elements. Let $Y(D)$ be the column vector consisting of the subscripts corresponding to the true category of D in the set of labels, then the prediction accuracy a_k of T_k for D is as follows:

$$a_k = Acc(\text{Predict}(T_k, D), Y(D)) \tag{6}$$

The decision weight that T_k occupies in the RF can be assigned based on the size of a_k. The higher the subtree prediction accuracy, the larger the decision weight it possesses, and the lower the subtree prediction accuracy, the smaller the decision weight it possesses. And according to a previous study by Zhao [24], it was found that the integrated learning classifier outperforms the single individual learner provided that non-negative weights are used. Therefore this paper defines the decision weight W_k occupied by T_k in RF as follows:

$$W_k = \frac{e^{a_k}}{\sum_{i=1}^{t} e^{a_i}} \tag{7}$$

$$\sum_{i=1}^{t} W_i = W_1 + W_2 + \cdots + W_t = \frac{e^{a_1}}{\sum_{i=1}^{t} e^{a_i}} + \frac{e^{a_2}}{\sum_{i=1}^{t} e^{a_i}} + \cdots + \frac{e^{a_t}}{\sum_{i=1}^{t} e^{a_i}} = 1 \tag{8}$$

According to the range of values of the prediction accuracy a_k of T_k, it is easy to obtain $W_k > 0$, and from Equation (8), it is easy to prove $\sum_{i=1}^{t} W_i = 1$, that is, it can be guaranteed that the sum of the predicted category probabilities of the samples of RF is still 1. Combining the above equations, the new category probability matrix computed by T_k in RF after learning the subtree decision weights on D is obtained as follows:

$$\text{Probability}_{new}(T_k, D) = \begin{pmatrix} p_{11} & p_{12} & p_{13} & \cdots & p_{1c} \\ p_{21} & p_{22} & p_{23} & \cdots & p_{2c} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{m1} & p_{m2} & p_{m3} & \cdots & p_{mc} \end{pmatrix} * W_k \tag{9}$$

Subsequently, the new category probability matrices obtained after weighting all subtrees in the RF are summed to obtain the following category probability matrix predicted by the RF on D:

$$\text{Probability}(F, D) = \sum_{i=1}^{t} \text{Probability}(T_i, D) * W_i \tag{10}$$

As a result, the category prediction result of the i-th training sample in the training set can be obtained based on the category label corresponding to the maximum value in the i-th row of the category probability matrix of the RF, and when there are multiple identical maximum values in the i-th row, the category label corresponding to one of the maximum values can be randomly selected as the prediction result of the RF for the i-th training sample in D.

Eventually, the training learns the following predictions for each subtree decision weight RF on S:

$$\text{Probability}(F, S) = \sum_{i=1}^{t} \text{Probability}(T_i, S) * W_i \tag{11}$$

4. **Intrusion risk detection model for power systems based on deep neural networks and integrated learning.**

4.1. **CNN-based local feature extraction for power system network traffic.** To improve the detection effect of the power system intrusion risk detection model, this article designs an invasion risk detection model relied on deep neural network and integrated learning, as shown in Figure 3. The input level of the method is the network traffic data of the power system. After the input level, the local features of the network traffic are extracted by CNN, and the extracted data are fed into the BiLSTM network for global learning to extract the two-way long-term temporal features, and the attention scheme is used to reevaluate the input characteristics, so that new weights are given to the traffic characteristics to suppress the useless information. Finally, the optimized RF integrated learning method outputs the detection results.

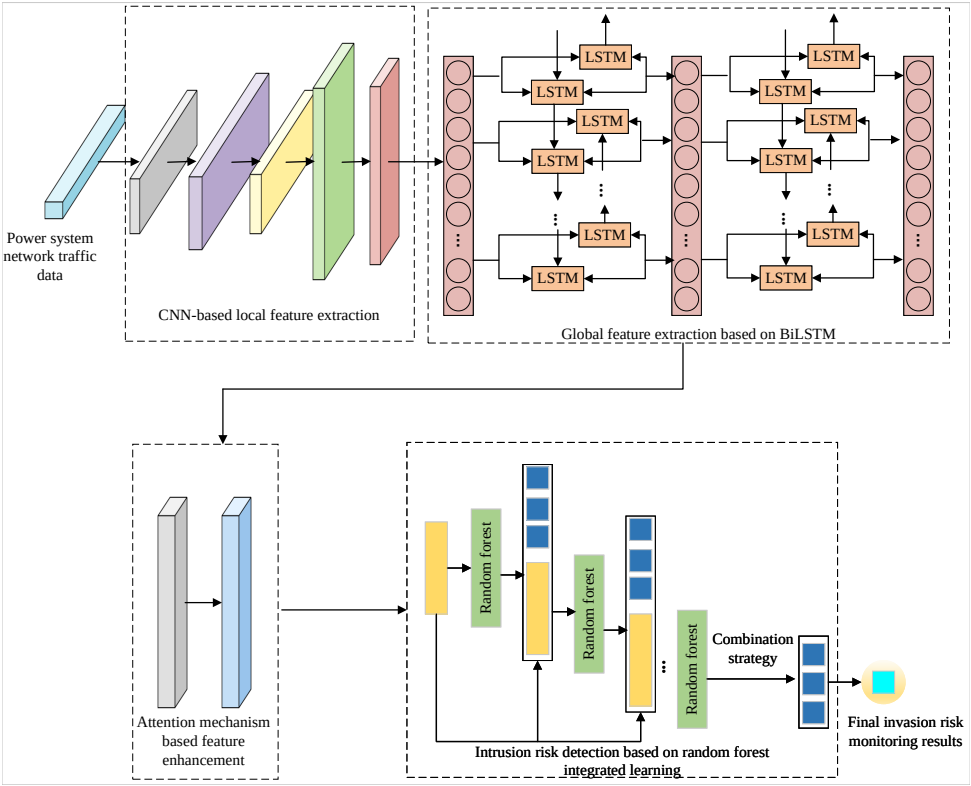

Figure 3. Designed intrusion risk detection model for power systems

In real power systems, there are various types of network traffic data, such as TCP, UDP, ICMP, etc., which have different characteristics and behavioral patterns [25]. The power system network traffic data is preprocessed by handling operations such as duplicate columns, missing values, data imbalance, etc., and then the preprocessed network data x is normalized to eliminate the proportional differences between the data, to accelerate the convergence of the algorithms, and to enhance the performance and prediction capability of the model, as shown below:

$$x^* = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{12}$$

Suppose $X = \{x_0, x_1, x_2, \ldots, x_n\}$ is a sequence of normalized network traffic data, and the elements in X denote attributes such as IP address, MAC address and memory address. Let the convolution kernel size be $2P + 1$, convolution kernel weight $W = \{w_{-p}, w_{-p+1}, \ldots, w_0, \ldots, w_{p-1}, w_p\}$. The new feature sequence generated by convolution

is $\chi = \{\chi_0, \chi_1, \chi_2, \ldots, \chi_n\}$, as shown below:

$$\chi_j = \sum_{k=-p}^{p} x_{j+k} w_k + b \tag{13}$$

where j is the position of the element in the new sequence of features and b is the bias term.

A is the i-th one-dimensional feature sequence containing k features, n one-dimensional feature sequences form a traffic cluster as input, denoted as $\chi_{1:n} = \chi_1 \oplus \chi_2 \oplus \cdots \oplus \chi_n$, where $\oplus$ is a sequence splicer, e.g., $\chi_{i:i+j}$ refers to the splicing of a one-dimensional sequence of features $\chi_i, \chi_{i+1}, \ldots, \chi_{i+j}$. The convolution operation consists of a filter $u \in R^{hk}$, which is used to convolve a sequence of h one-dimensional features to obtain a new feature c_i:

$$c_i = f(u \cdot \chi_{i:i+h-1} + b) \tag{14}$$

where f is the Leaky ReLU function that convolves the feature sequence $\{\chi_{1:h}, \chi_{2:h+1}, \ldots, \chi_{n-h+1:n}\}$ extracted from each sliding window in the traffic cluster by this filter to generate the final feature sequence c, where $c \in R^{n-h+1}$:

$$c = [c_1, c_2, \ldots, c_{n-h+1}] \tag{15}$$

4.2. **Global feature extraction of power system network traffic based on BiLSTM.** In power systems, attackers usually carry out attack behaviors over a sustained period of time rather than in a single instant. To better detect the risk of intrusion, these attacks can be identified by analyzing the correlations and changes in the time series before and after the network data. In this paper, BiLSTM [26] is utilized to learn features from the forward and backward bi-directional information of sequence data to obtain global feature information.

The feature sequence $c = [c_1, c_2, \ldots, c_{n-h+1}]$ is used as input to BiLSTM, in which the two LSTM networks are each incorporated into a unified structure. Unlike one-way LSTM networks that only use forward information to predict the output of the next state, BiLSTM utilizes the forward and backward embedding sequences of the two networks for joint processing. By integrating the information from the two LSTM networks, BiLSTM achieves the prediction of the output at subsequent times, i.e., the final output is a synthesis of the outputs of the two LSTM networks. The formulas of BiLSTM are shown in Equation (16), Equation (17) and Equation (18):

$$\vec{h}_t = \overrightarrow{LSTM}(\chi_t), t \in [1, T] \tag{16}$$

$$\overleftarrow{h}_t = \overleftarrow{LSTM}(\chi_t), t \in [T, 1] \tag{17}$$

$$h_t = [\vec{h}_t, \overleftarrow{h}_t] \tag{18}$$

where $\vec{h}_t$ and $\overleftarrow{h}_t$ represent the forward embedding sequence and the reverse embedding sequence, respectively, and for $\overrightarrow{LSTM}$, the input gate consists of two parts, the first part uses the Sigmoid activation function to obtain the output i_t, which represents the importance of the new information, and the second part uses the tanh activation function, which has the output a_t, which represents the candidate state vector to be updated:

$$i_t = \sigma(W_i \vec{h}_{t-1} + U_i \chi_i + b_i) \tag{19}$$

$$a_t = \tanh(W_a \vec{h}_{t-1} + U_a \chi_i + b_a) \tag{20}$$

where W_i, U_i, W_a, U_a are weights, b_i, b_a are biases, and h_{t-1} is the hidden state.

The information stored in the state memory unit is then flexibly adjusted by updating the gate, retaining useful information and filtering out useless information, as shown below:

$$C_t = C_{t-1} \odot f_t + i_t \odot a_t \tag{21}$$

where $\odot$ is the Hadamard product.

The output gate is responsible for determining how much cell state information should be output for the hidden state of the current time step and is calculated as below:

$$o_t = \sigma(W_o \vec{h}_{t-1} + U_o \chi_t + b_o) \tag{22}$$

$$\tilde{h}_t = o_t \odot \tanh(C_t) \tag{23}$$

Similarly, the inverse embedding operation by LSTM can be obtained as $\tilde{h}_t$. Finally, the global feature extracted by BiLSTM containing time information can be obtained as F:

$$F = a_t o_t + i_t o_t \sigma(U_i \chi_t + W_i(h_t + b_c)) \tag{24}$$

4.3. **Intrusion risk detection based on random forest integrated learning.** After obtaining the global characteristics of the power system network traffic F, the attention mechanism is utilized to enhance the critical features and ignore the redundant features, assuming that $F = \{f_1, f_2, \ldots, f_n\}$, some vector in the target value is q. The attention distribution is the conditional probability distribution of selecting the i-th input vector given X and q:

$$\alpha_i = P(i|X, q) = softmax(s(f_i, q)) = \frac{e^{s(f_i,q)}}{\sum_{j=1}^{n} e^{s(f_j,q)}} \tag{25}$$

where α_i is the attention distribution normalized by the softmax function, and $s(f_i, q)$ is the scoring function used to calculate the degree of similarity or correlation between f_i and q.

The cascaded WORF integration learning algorithm is utilized to integrate the predictions of each base classifier to output the final detection results, and the final attention value can be obtained by weighting and summing F according to the above attention distribution calculation results, so as to get the enhanced global feature as F'.

$$Attention(F, q) = \sum_{i=1}^{n} \alpha_i \cdot f_i \tag{26}$$

For the real issue of intrusion risk detection, it can be abstracted as a classification task, and the classification prediction is performed using the voting method [27] at the last level of the cascading WORF. Learner h_i predicts a labeling from the set of category markers $\{l_1, l_2, \ldots, l_N\}$, and represents the predicted output of h_i on F' as an N-dimensional vector $(h_i^1(F'); h_i^2(F'); \ldots; h_i^N(F'))$, where $h_i^j(F')$ is the output of h_i on the category labeling c_j. In the power network intrusion risk detection task with high reliability needs, the absolute majority voting approach is adopted, and if a mark receives more than half of the votes, the prediction is made for that mark, otherwise the forecasting is rejected. However, if the testing task needs a prediction must be offered, the absolute majority voting approach degenerates into the relative majority voting approach. The calculations for the absolute and relative majority voting methods are shown below:

$$H(F') = \begin{cases} l_j, & \text{if } \sum_{i=1}^{T} h_i^j(F') > 0.5 \sum_{k=1}^{N} \sum_{i=1}^{T} h_i^k(F') \\ reject, & \text{otherwise} \end{cases} \tag{27}$$

$$H(F') = l_{\arg\max} \sum_{i=1}^{T} h_i^j(F') \tag{28}$$

5. Experiment and result analysis.

5.1. Analysis of power system intrusion risk detection results.

This paper uses the publicly available power network dataset CIC-IDS-2017 [28] to conduct experiments. This dataset can effectively simulate the real network environment of the power system, and provides an experimental basis for the research of power system intrusion risk monitoring. The dataset was collected as of 5 p.m. on Friday, July 7, 2017, for a total of five days. Mondays contain only normal traffic, while Tuesday through Friday generate attack traffic during a fixed period of time. The dataset is feature-rich, with each piece of data consisting of 79 feature attributes and 12 security risk category labels, and the realized attacks include brute force FTP, denial-of-service attacks DOS, and Web attacks. The experiments were completed under Windows 10 operating system to train and test the model, and the proposed intrusion risk detection model was programmed using the Python libraries TensorFlow and Keras. The learning rate in the model is 0.01, the dropout rate is 0.01, the optimizer is Adam, and Epoch is set to 100.

The detection accuracy of the proposed model CNN-ORF for different categories of risks is shown in Figure 4, BENIGN, PortScan, FTP-Patator, SSH-Patator, Bot, Brute Force, Attack-XSS, Infiltration, SQL Injection and Heartbleed all have detection accuracy rates above 90%, and these risks all have unique characteristics that make them easier to detect compared to other risks. However, the detection accuracy of DoS Hulk, DDoS, DoS GoldenEye, DoS Slowloris, and DoS Slowhttptest is relatively low, because these five types of risks are DOS attacks, with similar attack modes and characteristics, and thus the detection accuracy is not as good as that of the other ten risks. However, the overall detection accuracy of the proposed model is above 90%, which verifies the efficiency of the suggested model.

The detection time overhead for various classes of risks is shown in Figure 5, the detection time for PortScan, SQL Injection, and Heartbleed is less than 1s. SQL Injection has the lowest detection time and is the easiest to detect because the attacker inserts or "injects" malicious SQL code into the application to gain unauthorized access. The detection times of DoS Hulk, DDoS, DoS GoldenEye, DoS Slowloris, and DoS Slowhttptest are relatively close to each other, fluctuating between $3.03 - 3.82s$, which is consistent with the above analysis of the detection accuracy of each risk category, and further validates the effectuality of the designed model.

5.2. Intrusion risk detection performance comparison and analysis.

To better evaluate the detection performance of the designed models, this article uses the accuracy, F1, false positive rate (FPR), underreporting rate (UR), detection rate (DR), and complete rate (CR) in order to compare and analyze the LSTM-SVM, R-CNN, CNN-GRU, RF-HP, and PSO-XG models, as indicated in Table 1.

As can be seen from Table 1, LSTM-SVM has the worst performance in all categories, with Accuracy of 73.59%, F1 of 70.17%, FPR of 4.19%, UR of 0.59%, DR of 81.64% and CR of 91.59%. CNN-ORF has the best metrics, with Accuracy and F1 improved by $2.89\% - 24.66\%$ compared to LSTM-SVM, R-CNN, CNN-GRU, RF-HP and PSO-XG, FPR and UR reduced by $0.38\% - 3.04\%$ compared to LSTM-SVM, R-CNN, CNN-GRU, RF-HP and PSO-XG decreased by $0.38\% - 3.04\%$, and DR and CR improved by $0.51\% - 17.04\%$ compared to LSTM-SVM, R-CNN, CNN-GRU, RF-HP and PSO-XG. LSTM-SVM does not take into account the local features of the power network traffic

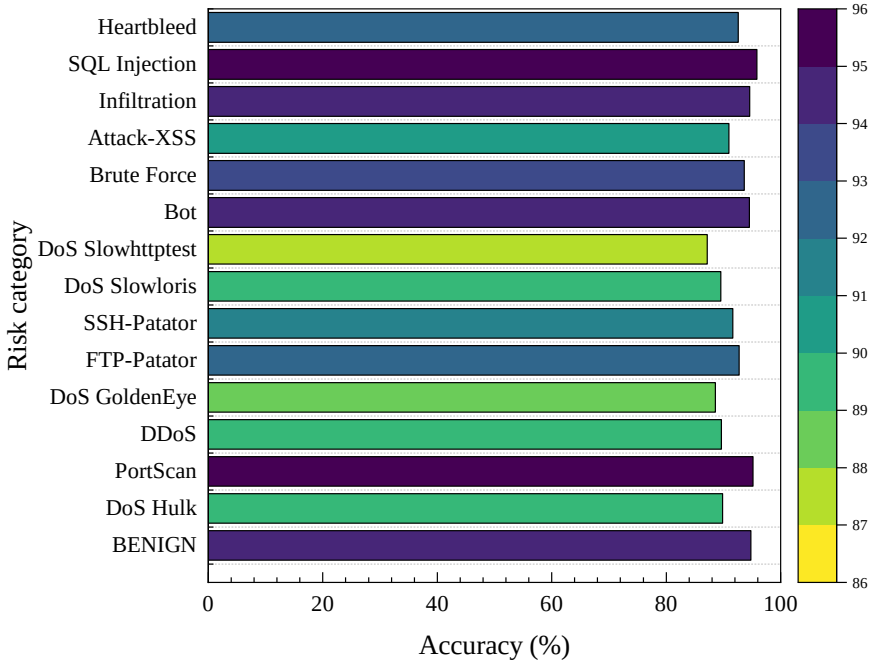

Figure 4. The detection accuracy of the proposed model for different categories of risks

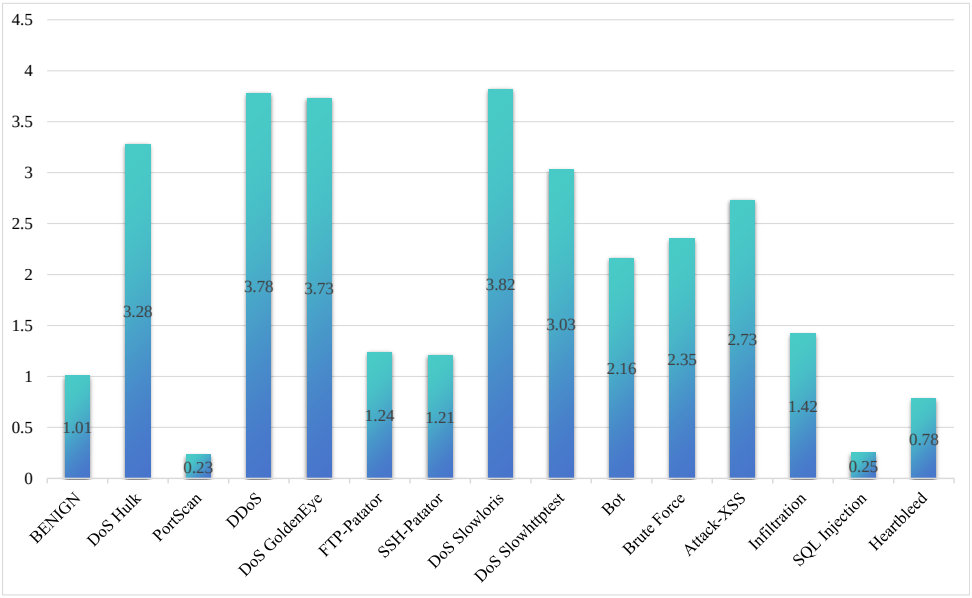

Figure 5. Detection time overhead for various classes of risk (s)

data and all the classifiers have high computational complexity, resulting in inefficient detection. Both R-CNN and CNN-GRU perform feature extraction of power network traffic data through CNN, neither considering temporal features nor augmenting critical features. RF-HP integrates multiple classifiers such as SVM, DT, etc. through traditional RF algorithms, but feature extraction is not sufficient. PSO-XG optimizes the XGBoost integrated learning algorithm by PSO, which improves the detection performance to a

Table 1. Detection performance metrics of different models

Model	Accuracy/%	F1/%	FPR/%	UR/%	DR/%	CR/%
LSTM-SVM	73.59	70.17	4.19	0.59	81.64	91.59
R-CNN	77.63	74.14	3.53	0.51	86.39	94.82
CNN–GRU	81.65	83.41	2.85	0.46	89.51	96.41
RF-HP	87.24	88.91	2.30	0.42	94.58	97.53
PSO-XG	90.52	88.12	1.93	0.30	97.53	99.12
CNN-ORF	93.41	94.83	1.15	0.21	98.68	99.63

certain extent, but it does not remove the redundant features, so the detection efficiency is not as good as CNN-ORF.

The PR curves of different models are shown in Figure 6. In the performance comparison of PR curves, if the curve of model LSTM-SVM is completely above the curve of model R-CNN, then there is no doubt that the detection performance of LSTM-SVM is better than that of R-CNN. As can be seen from the figure, the PR curves of CNN-ORF are all above the other five models, further verifying the superiority of its detection performance. CNN-ORF not only utilizes CNN to extract local features of the power system, but also utilizes BiLSTM for global feature extraction containing temporal features. In addition, CNN-ORF enhances the importance features using the attention mechanism and optimizes the RF integration learning algorithm, which greatly improves the detection performance.

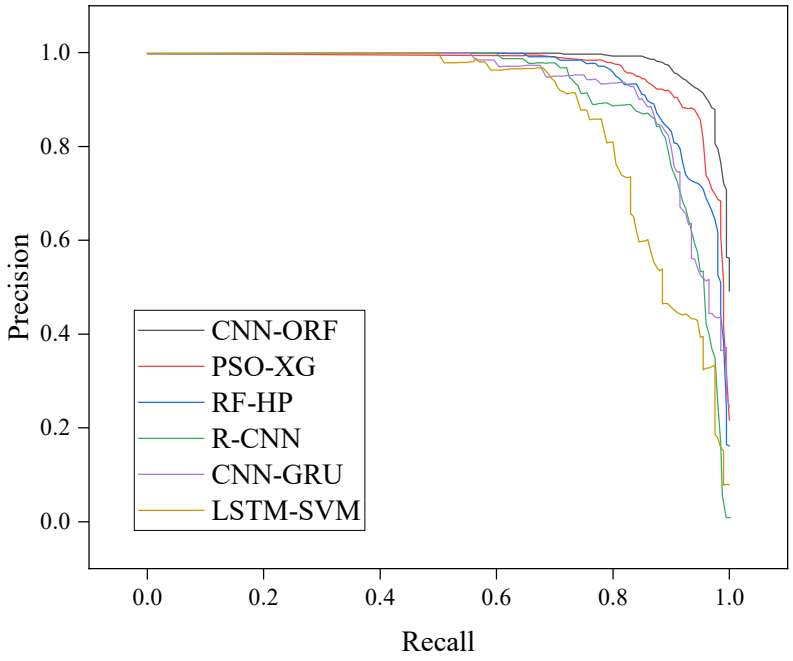

Figure 6. Comparison of PR curves for different models

6. **Conclusion.** With the large-scale application of advanced technologies in the new power system, the safety of the power system is bound to become more complex and the hazards caused by it more worrying. This paper designs a power system intrusion risk detection model based on deep neural network and integrated learning for the existing power

system intrusion detection model with difficult feature extraction and unsatisfactory detection effect. Firstly, the combination strategy of RF integrated learning is optimized based on the idea of exponential weighting to re-quantify the decision weighting of base classifiers in random forests, and to reduce the skewed influence of base classifiers with lower prediction accuracies on the prediction results of the whole integrated learning. Then CNN is utilized for local feature extraction of power system network traffic, and the localized features are fed into BiLSTM network for global learning, and global features containing temporal attributes are extracted by capturing the sequential dependencies between features before and after, and the attention mechanism is introduced to ignore irrelevant features. Finally, the cascaded RF integrated learning algorithm is utilized to integrate the predictions of each base classifier and output the final detection results. The experiments indicate that the suggested model has a high detection accuracy in the power system environment, which provides a strong support to ensure the safe and stable operation of the power system. The detection efficiency of the suggested model has been enhanced to a certain extent in the comparative experiment, but there is still room for improvement. In the future, this research will focus on the lightweight implementation of the model and the optimization of the model in the GPU environment.

REFERENCES

[1] M. Zadsar, A. Abazari, A. Ameli, J. Yan, and M. Ghafouri, "Prevention and detection of coordinated false data injection attacks on integrated power and gas systems," *IEEE Transactions on Power Systems*, vol. 38, no. 5, pp. 4252-4268, 2022.

[2] T.-Y. Wu, H. Li, S. Kumari, and C.-M. Chen, "A Spectral Convolutional Neural Network Model Based on Adaptive Fick's Law for Hyperspectral Image Classification," *Computers, Materials & Continua*, vol. 79, no. 1, pp. 19-46, 2024.

[3] P. Radoglou Grammatikis, P. Sarigiannidis, G. Efstathopoulos, and E. Panaousis, "ARIES: A novel multivariate intrusion detection system for smart grid," *Sensors*, vol. 20, no. 18, 5305, 2020.

[4] M. Panthi, and T. K. Das, "Intelligent intrusion detection scheme for smart power-grid using optimized ensemble learning on selected features," *International Journal of Critical Infrastructure Protection*, vol. 39, 100567, 2022.

[5] S. Pan, T. Morris, and U. Adhikari, "Developing a hybrid intrusion detection system using data mining for power systems," *IEEE Transactions on Smart Grid*, vol. 6, no. 6, pp. 3104-3113, 2015.

[6] S. Umar, and M. Felemban, "Rule-based detection of false data injections attacks against optimal power flow in power systems," *Sensors*, vol. 21, no. 7, 2478, 2021.

[7] D. Upadhyay, J. Manero, M. Zaman, and S. Sampalli, "Gradient boosting feature selection with machine learning classifiers for intrusion detection on power grids," *IEEE Transactions on Network and Service Management*, vol. 18, no. 1, pp. 1104-1116, 2020.

[8] T. Rose, K. Kifayat, S. Abbas, and M. Asim, "A hybrid anomaly-based intrusion detection system to improve time complexity in the Internet of Energy environment," *Journal of Parallel and Distributed Computing*, vol. 145, pp. 124-139, 2020.

[9] Z. Azam, M. M. Islam, and M. N. Huda, "Comparative analysis of intrusion detection systems and machine learning based model analysis through decision tree," *IEEE Access*, vol. 11, pp. 80348 – 80391, 2023.

[10] P. Radoglou Grammatikis, P. Sarigiannidis, G. Efstathopoulos, and E. Panaousis, "ARIES: A novel multivariate intrusion detection system for smart grid," *Sensors*, vol. 20, no. 18, 5305, 2020.

[11] C. Song, Y. Sun, G. Han, and J. J. Rodrigues, "Intrusion detection based on hybrid classifiers for smart grid," *Computers & Electrical Engineering*, vol. 93, 107212, 2021.

[12] F. Gao, S. Ji, J. Guo, Q. Li, Y. Ji, Y. Liu, S. Feng, H. Wei, N. Wang, and B. Yang, "ID-Net: an improved mask R-CNN model for intrusion detection under power grid surveillance," *Neural Computing and Applications*, vol. 33, pp. 9241-9257, 2021.

[13] F. Zhai, T. Yang, H. Chen, B. He, and S. Li, "Intrusion detection method based on CNN–GRU–FL in a smart grid environment," *Electronics*, vol. 12, no. 5, 1164, 2023.

[14] G. Zhang, J. Li, O. Bamisile, Y. Xing, D. Cao, and Q. Huang, "Identification and classification for multiple cyber attacks in power grids based on the deep capsule CNN," *Engineering Applications of Artificial Intelligence*, vol. 126, 106771, 2023.

[15] Z. Chen, N. He, Y. Huang, W. T. Qin, X. Liu, and L. Li, "Integration of a deep learning classifier with a random forest approach for predicting malonylation sites," *Genomics, Proteomics and Bioinformatics*, vol. 16, no. 6, pp. 451-459, 2018.

[16] J. Sun, G. Wang, G. He, D. Pu, W. Jiang, T. Li, and X. Niu, "Study on the water body extraction using GF-1 data based on adaboost integrated learning algorithm," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 42, pp. 641-648, 2020.

[17] J. Kang, X. Guo, L. Fang, X. Wang, and Z. Fan, "Integration of Internet search data to predict tourism trends using spatial-temporal XGBoost composite model," *International Journal of Geographical Information Science*, vol. 36, no. 2, pp. 236-252, 2022.

[18] N. Zhu, C. Zhu, L. Zhou, Y. Zhu, and X. Zhang, "Optimization of the random forest hyperparameters for power industrial control systems intrusion detection using an improved grid search algorithm," *Applied Sciences*, vol. 12, no. 20, 10456, 2022.

[19] M. Panthi, and T. K. Das, "Intelligent intrusion detection scheme for smart power-grid using optimized ensemble learning on selected features," *International Journal of Critical Infrastructure Protection*, vol. 39, 100567, 2022.

[20] I. Singh, N. Kumar, K. Srinivasa, T. Sharma, V. Kumar, and S. Singhal, "Database intrusion detection using role and user behavior based risk assessment," *Journal of Information Security and Applications*, vol. 55, 102654, 2020.

[21] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A survey of convolutional neural networks: analysis, applications, and prospects," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 12, pp. 6999-7019, 2021.

[22] Q. Gu, J. Tian, X. Li, and S. Jiang, "A novel Random Forest integrated model for imbalanced data classification problem," *Knowledge-Based Systems*, vol. 250, 109050, 2022.

[23] P. A. A. Resende, and A. C. Drummond, "A survey of random forest based methods for intrusion detection systems," *ACM Computing Surveys (CSUR)*, vol. 51, no. 3, pp. 1-36, 2018.

[24] H. Zhao, and S. Ram, "Entity identification for heterogeneous database integration—a multiple classifier system approach and empirical evaluation," *Information Systems*, vol. 30, no. 2, pp. 119-132, 2005.

[25] I. Kotenko, I. Saenko, O. Lauta, and A. Kribel, "An approach to detecting cyber attacks against smart power grids based on the analysis of network traffic self-similarity," *Energies*, vol. 13, no. 19, pp. 5031, 2020.

[26] W. Liu, W. Jing, and Y. Li, "Incorporating feature representation into BiLSTM for deceptive review detection," *Computing*, vol. 102, no. 3, pp. 701-715, 2020.

[27] C. Cornelio, M. Donini, A. Loreggia, M. S. Pini, and F. Rossi, "Voting with random classifiers (VORACE): theoretical and experimental analysis," *Autonomous Agents and Multi-Agent Systems*, vol. 35, no. 2, 22, 2021.

[28] Z. Pelletier, and M. Abualkibash, "Evaluating the CIC IDS-2017 dataset using machine learning methods and creating multiple predictive models in the statistical computing language R," *Science*, vol. 5, no. 2, pp. 187-191, 2020.