DEFECT PREDICTION METHOD FOR POWER TRANSMISSION EQUIPMENT BASED ON GREY WOLF OPTIMIZATION ALGORITHM AND LSTM-HNA

Zhennan YANG[1], Jie ZHANG[2]*, Ke ZHANG[3], Youqiang SUN[2], Wenli HUANG[3], Le ZOU[1]

Predicting defects in power transmission equipment is crucial for ensuring the stable operation of the power grid. However, existing methods suffer from shortcomings such as ignoring temporal features, susceptibility to irrelevant feature interference, and lack of hyperparameter optimization. To address this, this paper proposes a novel method for defect prediction in power transmission equipment that combines the Grey Wolf Optimization (GWO) algorithm with Long Short-Term Memory networks and an attention mechanism. This method utilizes LSTM networks to extract temporal feature information, innovatively designs a Hidden-layer Neuron Attention module (HNA) to reduce the impact of irrelevant features, incorporates the Grey Wolf Optimization algorithm for automatic hyperparameter tuning, and proposes a joint training strategy for GWO and LSTM-HNA to improve efficiency. Extensive experiments validate the effectiveness of the proposed method, achieving a high accuracy of 97.37% in defect prediction tasks for power transmission equipment. Precision, recall, F1 score, and other metrics outperform other methods, providing a new approach to enhancing the monitoring and fault prevention capabilities of power transmission equipment.

Keywords: Power Transmission Equipment; Defect Prediction; Grey Wolf Optimization Algorithm; Attention Mechanism; Long Short-Term Memory Network

1. Introduction

To meet the growing demand for electricity, an increasing number of large substations are being constructed and operated [1]. Substations, as crucial nodes in the transmission system, play a vital role in ensuring the overall reliability of the power system. Therefore, the timely detection and prediction of potential defects in substation equipment, along with the implementation of effective maintenance and preventive measures, are of significant importance for ensuring the safe operation of the grid, reducing risks, and maintaining cost-effectiveness.

Currently, with the aging of equipment intensifying, the risks of defects and failures are increasing annually, and traditional manual inspections and periodic

[1] School of Artificial Intelligence and Big Data, Hefei University, Hefei 230601, China
[2] Institute of Intelligent Machine, Hefei Institute of Physical Science, Chinese Academy of Sciences, Hefei 230031, China
[3] Anhui NARI Jiyuan Power Grid Technology Co. Ltd., Hefei 230088, China
* Corresponding author's e-mail: zhangjie@iim.ac.cn

maintenance are no longer sufficient to meet the requirements of real-time monitoring. Consequently, the scientific and efficient maintenance of substation equipment has become an urgent issue. With the continuous development of artificial intelligence (AI) technology, attempts have been made to use AI techniques to address this challenge. One such approach is the use of AI-based techniques for predicting defects in substation equipment, [2]-[5] which involves extracting patterns from historical data accumulated during the operation of the equipment. By identifying characteristic features in the monitoring data before defects occur, timely warnings can be issued when similar data patterns reappear, enabling early detection and maintenance.

Existing research on substation equipment defect prediction includes methods such as [6] SMOTE-XGBoost-based transformer defect prediction, [7]A transformer fault diagnosis method based on Dissolved Gas Analysis (DGA) is proposed. This method combines the ReliefF feature weighting method and the HPO-SVM model, improving the accuracy of fault diagnosis, [8]autoencoder neural network-based grid fault prediction algorithms, [9] knowledge graph-based power transformer fault prediction methods, [10] least squares support vector machine Bayesian network decision tree (LSSVM-BNDT) defect diagnosis methods, and [11] a novel support vector machine multi-classification strategy for power transformer fault diagnosis. These methods primarily address issues such as missing or imbalanced datasets and, through data processing and model optimization, have somewhat improved prediction accuracy. However, these traditional methods neglect the inherent temporal dependencies within the data, failing to fully exploit crucial temporal information such as time trends and periodic patterns. Instead, they treat data simply as static feature vectors, limiting further improvements in predictive performance.

To capture temporal features, some researchers have proposed time series modeling methods, including [12] transformer fault prediction using Long Short-Term Memory networks (LSTM), [13] support vector machine-based power grid fault diagnosis methods, and [14] machine learning-based regression classification joint solutions for predicting power equipment defects, faults, and their occurrence times. Although these methods consider time series information, they inefficiently discern the importance of input features during the model training stage, resulting in the inclusion of redundant and non-critical features that impact generalization and prediction accuracy.

Addressing the aforementioned issues, this paper proposes a power equipment defect prediction method based on the Grey Wolf Optimization algorithm (GWO) combined with Long Short-Term Memory networks and an Attention Mechanism. This method: 1) effectively utilizes temporal information in defect data through the use of the LSTM model, improving defect prediction accuracy; 2) introduces the Hidden-layer Neural Attention module (HNA) to reduce

the impact of redundant features on model performance; 3) incorporates the GWO algorithm for automatic hyperparameter optimization; and 4) proposes a joint training mechanism to enhance model training efficiency. This approach improves the accuracy and efficiency of substation equipment defect prediction.

2. Materials and Methods

This section introduces the data used in this work and the details of the proposed model.

2.1 Datasets

In this study, the data acquisition encompasses the asset register, operational monitoring, and defect records of a specific substation. The operational monitoring data includes oil chromatography, top oil temperature, core grounding current, and power supply load data, among others. These data are collected in real-time by the online monitoring devices installed in the substation and are aggregated and organized by a data monitoring platform. The process of collecting online monitoring data is depicted in Fig. 1.

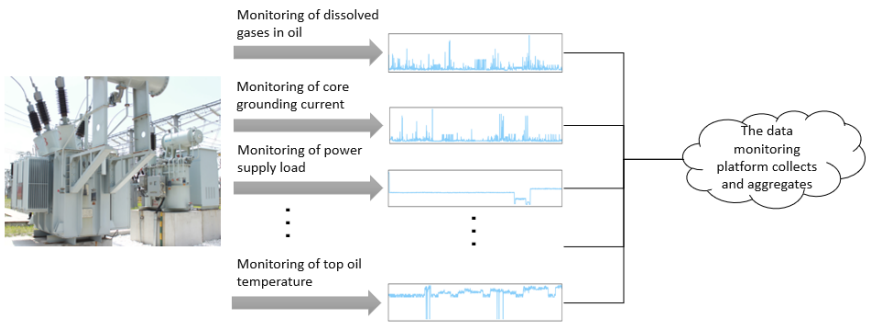

Fig. 1. The data acquisition process for online monitoring

2.2 Data Preprocessing

Before training the model, we preprocessed various types of collected data to construct a dataset suitable for training. These data include ledger data such as manufacturer, voltage level, and months in operation, as well as operational monitoring data such as power load, temperature, micro-water, and defect data.

2.2.1 Data Cleaning

In the data cleaning stage, we first identify and analyze outliers and missing values for numerical data. For outliers, we replace them with the mean value; for missing values, we fill them in. For non-numerical data, we analyze and process missing and erroneous values, typically choosing to delete them.

2.2.2 Feature Vectorization

Since string-type data cannot be directly used for neural network training, we need to convert them into vector form. Although label encoding and one-hot encoding

are mainstream feature vectorization methods, they each have their limitations. Label encoding typically requires clear size relationships between categories, while one-hot encoding can lead to a sharp increase in feature dimensions when there are many categories. Therefore, we use target encoding to vectorize categorical variables. This method calculates the mean label of the feature category and replaces the category value with this mean value, thereby avoiding increasing data dimensions and enhancing the interpretability of the encoding.

2.3 LSTM model

Long Short-Term Memory (LSTM) is an improved type of Recurrent Neural Networks (RNN) known for its memory capabilities [15][16]. Compared to traditional RNNs [17], LSTM exhibits excellent performance in handling long time series problems. Its key lies in the carefully designed gate mechanisms and state transition methods, which enable more effective capturing and utilization of information embedded in long-term temporal dependencies, thus avoiding the vanishing or exploding gradient problem and significantly enhancing the modeling ability for long sequences of data. The LSTM neuron structure is illustrated in Fig. 2.

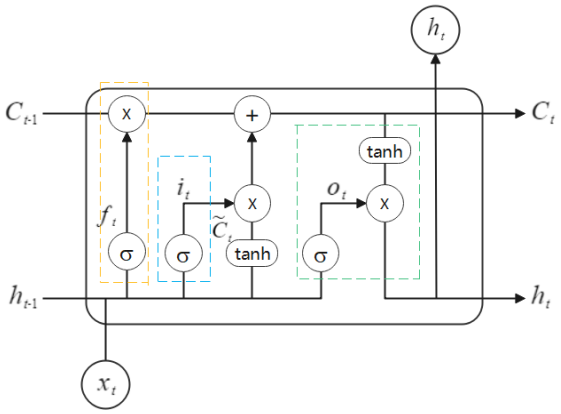

Fig. 2. LSTM neuron structure

The core design idea of LSTM networks is to selectively retain and forget information through carefully designed gate mechanisms to capture long-term temporal dependencies. Specifically:

The forget gate determines which information from the previous time step's cell state needs to be forgotten, and its calculation formula is:

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

Where f_t is the activation value vector of the forget gate, W_f and b_f are the weight matrix and bias, h_{t-1} is the previous time step's hidden state, and x_t is the current input.

The input gate controls the extraction and updating of useful information from the current input and the memory cell vector:

$$i_t = \sigma(W_i \cdot [h_{t-1} + x_t] + b_i) \quad (2)$$

$$\widetilde{C_t} = \tanh(W_c \cdot [h_{t-1} + x_t] + b_c) \quad (3)$$

where i_t is the activation value vector of the input gate, $\widetilde{C_t}$ is the new candidate memory cell vector.

The cell state C_t is the core of LSTM, responsible for transmitting and storing long-term state information, and its update rule is:

$$C_t = f_t \odot C_{t-1} + i_t \odot \widetilde{C_t} \quad (4)$$

Finally, the output gate controls the output hidden state h_t based on the cell state and the current input:

$$o_t = \sigma(W_o \cdot [h_{t-1} + x_t] + b_o) \quad (5)$$

$$h_t = o_t \odot \tanh(C_t) \quad (6)$$

Through the carefully designed gate mechanisms described above, LSTM can efficiently learn and extract key temporal patterns and trend information from long-term input sequences, significantly improving its ability to model long-term dependencies.

2.4 The model based on Grey Wolf Optimization and LSTM-HNA

We have designed a LSTM-HNA (Long Short-Term Memory with Hidden Layer Neuron Attention) model for defect prediction of power substation equipment based on the Grey Wolf Optimization Algorithm. This model consists of two main modules: the hyperparameter optimization module and the main network module. The framework of the model is shown in Fig. 3.

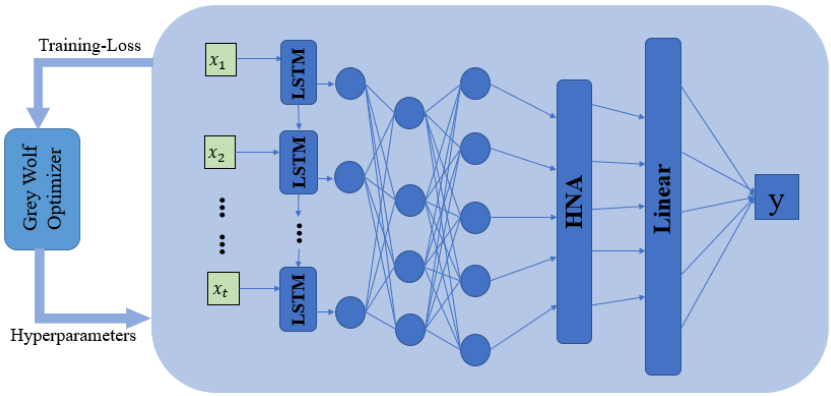

Fig. 3. Model framework

The hyperparameter optimization module utilizes the Grey Wolf Optimization Algorithm to automatically search for the optimal hyperparameters. In the main network module, to fully utilize the temporal information in the historical defect data of the equipment and improve prediction accuracy, we adopt the LSTM network. Additionally, to enhance model performance and reduce the impact of redundant features, we introduce a hidden layer neuron attention module. The workflow of the model is as follows: first, the defect data with temporal

information is input into the LSTM layer. Then, the hidden state of the LSTM layer is passed to the hidden layer neuron attention module. Finally, the model prediction result is output through a linear layer. Combining the memory characteristics of LSTM and the dynamic attention ability of the attention mechanism, our model can more effectively model and learn representations of sequence data.

2.5 The proposed Attention model

2.5.1 Attention mechanism

The attention mechanism [18]-[20] is an efficient technique that mimics the way the human brain processes problems, aiming to simplify the resolution process of complex issues. When faced with complex problems, humans selectively focus on different information elements, prioritizing the most crucial information for the current problem, thereby reducing interference from irrelevant information. Similarly, by applying the attention mechanism, we can filter the input device defect features, calculate the contribution of each input feature to the output results, and effectively reduce the impact of irrelevant features on the model's performance.

2.5.2 Hidden Layer Neuron Attention

The Hidden Layer Neuron Attention (HNA) module is a novel mechanism proposed by us to enhance the performance of neural network models. By introducing the attention mechanism between hidden layer neurons, HNA allows each neuron to dynamically focus on the output of other neurons. This adaptive weight adjustment enables the model to more effectively learn and utilize the information of the input data, thereby improving overall performance. The hidden layer neuron attention module design is depicted in Fig. 4.

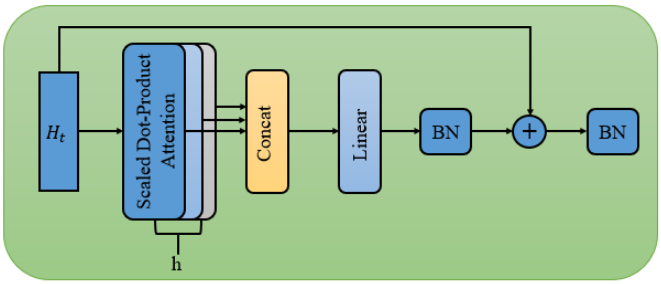

Fig. 4. HNA block

The workflow of the HNA module is as follows:

1) Multi-Head Attention Calculation: The hidden layer neurons are first processed through the multi-head attention mechanism, where the output of each attention head is concatenated and undergoes a linear transformation.

2) Layer Normalization: Subsequently, layer normalization is applied to the output from the previous step to stabilize the training process.

3) Residual Connection: The normalized output is then combined with the original hidden layer output through a residual connection to facilitate information flow and gradient propagation.

4) Final Output: Finally, layer normalization is applied again to obtain the final output of the HNA module.

Through this series of steps, the HNA module not only enhances the model's learning ability but also improves training stability and convergence speed, thereby enhancing the model's generalization ability.

2.6 Grey Wolf Optimizer for hyperparameter tuning

2.6.1 Grey Wolf Optimizer

The Grey Wolf Optimizer (GWO) [21]-[22] is an efficient intelligent optimization algorithm inspired by the hunting behavior of grey wolf packs. There are many similar swarm intelligence optimization algorithms that have also been widely applied to the hyperparameter tuning of neural networks [23][24]. Compared to other heuristic search algorithms, GWO has several significant advantages. Firstly, GWO demonstrates outstanding convergence performance, with not only fast convergence speed but also greater stability compared to algorithms like Particle Swarm Optimization. Secondly, GWO requires fewer parameter settings, making the algorithm easier to understand and implement, and consequently, the computation speed is faster. These characteristics have led to the widespread application of GWO in the field of parameter optimization.

In the Grey Wolf Optimizer algorithm, the social hierarchy of the wolf pack is divided into four levels: α (the leader), β (the deputy), δ (the scout), and ω (the ordinary wolf). The optimization process of the algorithm simulates the hunting strategy of grey wolves, where wolves of each rank play different roles in the search for solutions. The hierarchical division of the wolf pack is illustrated in Fig. 5.

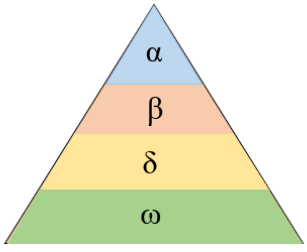

Fig. 5. Ranking in a wolf pack

During the search for the optimal solution, guidance is primarily provided by the α, β, and δ wolves, while the ω wolf follows. The hunting process of grey wolves can be divided into three main steps:

1) Encircling the prey: The process of grey wolves surrounding and gradually approaching the prey can be defined by the following formula:

$$D = \left| C \cdot X_p(t) - X(t) \right| \tag{7}$$

$$X(t+1) = X_p(t) - A \cdot D \tag{8}$$

Here, D represents the distance between the grey wolf and the prey, X_p and X are the position vectors of the prey and the grey wolf respectively. A and C are coefficient vectors calculated by the following formulas:

$$A = 2a \cdot r_1 - a \tag{9}$$

$$C = 2 \cdot r_2 \tag{10}$$

where a is the convergence factor, $r_1$and r_2 are random numbers in the interval [0,1].

2) Hunting: After the wolf pack discovers the prey, the β and δ wolves, led by the α wolf, surround the prey. The mathematical model for individual grey wolves tracking the prey's position is as follows:

$$\begin{aligned} D_\alpha &= |C_1 \cdot X_\alpha - X| \\ D_\beta &= |C_2 \cdot X_\beta - X| \\ D_\delta &= |C_3 \cdot X_\delta - X| \end{aligned} \tag{11}$$

Here, D_α, D_β, and D_δ represent the distances between the α, β, and δ wolves respectively, and the other individuals. X_α, X_β , and $X_\delta$represent the current positions of the α, β, and δ wolves respectively. C_1, C_2, and C_3 are random vectors. X_1, X_2, and X_3 define the step length and direction towards α, β, and δ respectively, and the final updated position of ω is defined as:

$$\begin{aligned} X_1 &= X_\alpha - A_1 \cdot (D_\alpha) \\ X_2 &= X_\beta - A_2 \cdot (D_\beta) \\ X_3 &= X_\delta - A_3 \cdot (D_\delta) \end{aligned} \tag{12}$$

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3} \tag{13}$$

3) Attacking the prey: When the prey stops moving, the grey wolves complete the hunting process by attacking to finally lock onto the optimal solution.

2.6.2 Hyperparameter Tuning

Hyperparameters play a crucial role in training neural network models, as they directly determine the performance and effectiveness of the training process. These hyperparameters include learning rate, number of neurons, number of epochs, batch size, etc., which are usually manually set based on experience, making it difficult to find an optimal set of hyperparameters.

To address this issue, we propose using the Grey Wolf Optimization Algorithm to automatically optimize the hyperparameters of neural network models. The main implementation steps are as follows:

Step 1: First, we set the learning rate, number of hidden layer neurons, number of epochs, and batch size as the optimization objectives of the population. Then, initialize the grey wolf population with N individuals and set the number of iterations as T.

Step 2: We use the loss value of model training as the fitness value of the population, determine the top three wolves with the best fitness values, and save the current optimal values.

Step 3: Calculate and update the parameters a, A, C as shown in equation (9) and equation (10).

Step 4: Calculate the positions of the grey wolves according to the position update formula (12) and formula (13).

Step 5: Repeat steps 2 to 4 until T iterations are completed. After each iteration, we compare the optimal values obtained to find a globally optimal set of hyperparameters.

Through this method, we can automatically and effectively find an optimal set of hyperparameters, thereby improving the training effectiveness and performance of neural network models. This approach has broad application prospects and is of great significance for the training of neural network models.

2.7 Training Strategy

For the model proposed by us, this study designed an innovative joint training strategy, which integrates the Grey Wolf Optimization Algorithm (GWO), Long Short-Term Memory (LSTM), and the attention mechanism of hidden layer neurons. By integrating the GWO algorithm into the main network, we can simultaneously search for the optimal hyperparameter combination during the model training process. As a result, after training, we not only obtain a model optimized for performance but also a set of precisely tuned optimal hyperparameters. The flowchart for the joint training strategy is shown in Fig. 6.

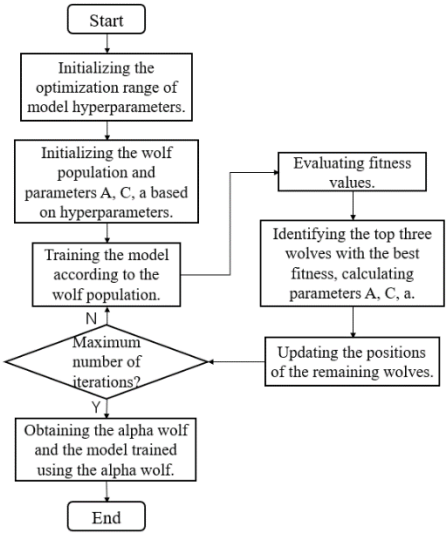

Fig. 6. Joint training flowchart

The specific process of the joint training strategy is as follows:

1) Initialization: Set the size of the grey wolf population and assign a set of hyperparameters to each grey wolf.

2) Model Training: Train the model using the hyperparameter combinations of each grey wolf individual and calculate the corresponding loss value.

3) Evaluation and Saving: Based on the loss value, identify and record the hyperparameter combinations of the top three grey wolves with the smallest losses.

4) Parameter Update: Update the hyperparameters of the grey wolf individuals using the GWO algorithm.

5) Iterative Optimization: Repeat the above process until the predetermined number of iterations is completed. Finally, select the hyperparameter combination with the smallest loss value and obtain the corresponding optimal model.

Through the above steps, our joint training strategy not only improves the training efficiency of the model but also ensures the optimization of hyperparameters. See the attached diagram for a detailed flowchart.

3 Experiment and Result Analysis

3.1 Experimental Data and Environment Setup

The data was collected from a substation in Anhui, primarily including three types: ledger data, online monitoring data, and defect data. The oil chromatography data records the monitoring values of various gases, such as hydrogen, methane, ethylene, etc., with a total of 19 features and 1,000 data sets. The specific features are shown in Table 1.

Tabel1

Feature variable statistics

Variable Name	Variable Type	Data Source
Device ID	String	Ledger Data
Manufacturer	String	Ledger Data
Voltage Level	String	Ledger Data
Months in Operation	Integer	Ledger Data
Power Load	Float	Monitoring Data
Core Grounding Current	Float	Monitoring Data
Top Oil Temperature	Float	Monitoring Data
Oil Chromatography	Float	Monitoring Data
Micro Water	Float	Monitoring Data
Temperature	Float	Monitoring Data
Weather	String	Monitoring Data
Month of Defect Occurrence	Integer	Defect Data
Defect Occurrence	String	Defect Data

The experimental environment is a Windows 10 operating system, with an NVIDIA GeForce MX150 GPU. The deep learning framework used is Pytorch, and the code is run in PyCharm. The data is split into training and testing sets in a 7:3 ratio for comparative experiments. The target variable is " Defect Occurrence ",

with values of 0 (no defect) and 1 (defect). The experiment uses the Adam optimizer with an initial learning rate of 0.001, and the loss function is the cross-entropy loss function.

3.2 Evaluation Metrics

In binary classification problems, each instance is classified as either positive or negative, resulting in four possible classification outcomes: True Positive (TP), False Negative (FN), False Positive (FP), and True Negative (TN). Specifically: True Positive (TP): Instances that are actually positive and correctly classified as positive. False Negative (FN): Instances that are actually positive but incorrectly classified as negative. False Positive (FP): Instances that are actually negative but incorrectly classified as positive. True Negative (TN): Instances that are actually negative and correctly classified as negative.

These classification outcomes form the basis for evaluating the performance of classification algorithms, with commonly used evaluation metrics including accuracy, recall, F1 score, and others. The specific evaluation metrics and their calculation methods are defined as follows:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{14}$$

$$Precision = \frac{TP}{TP+FP} \tag{15}$$

$$Recall = \frac{TP}{TP+FN} \tag{16}$$

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision \cdot Recall} \tag{17}$$

(1) ROC (Receiver Operating Characteristic) Curve

The ROC curve is an important tool for evaluating the performance of classification models. It depicts the model's ability to identify signals by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) at different thresholds. Each point on the ROC curve represents the model's response to signal stimuli at a specific threshold. Specifically horizontal axis (FPR) represents the rate at which the model incorrectly classifies negative instances as positive, also known as 1-specificity. vertical axis (TPR) represents the rate at which the model correctly identifies positive instances, also known as sensitivity. When the ROC curve is closer to the upper-left corner (0,1), the model's ability to identify signals is stronger, indicating better discrimination between positive and negative classes.

(2) AUC (Area Under the Curve)

The AUC value is the area under the ROC curve, providing a quantitative measure of the model's classification ability. The AUC value ranges from 0 to 1, where a higher AUC value closer to 1 indicates stronger classification ability. It signifies the model's higher capability to correctly differentiate between positive and negative classes across various thresholds.

3.3 Comparative Experiments

To validate the effectiveness of the GWO-LSTM-HNA network model in predicting defects in power substation equipment, extensive comparative experiments were conducted on the dataset in this study. The model was compared with various mainstream classification models, including Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), LightGBM, and Multilayer Perceptron (MLP). Throughout the experiments, we meticulously recorded the performance of each model on four key performance metrics: precision, recall, F1-score, and accuracy. The experimental results are presented in Table .

Table 2

Comparison experiments with traditional methods

Method	Precision	Recall	F1-score	Accuracy
SVM	73.74	75.16	78.98	85.15
DecisionTree	79.35	76.49	77.80	89.23
RandomForest	93.16	73.52	79.34	91.83
LightGBM	94.35	82.77	88.12	94.88
MLP	79.04	65.27	70.58	89.46
GWO-LSTM-HNA	**97.71**	**91.65**	**94.38**	**97.37**

As shown in Table , the GWO-LSTM-HNA model achieved an accuracy of 97.37%, which is 12.2% higher than the lowest-performing SVM model and 2.5% higher than the second-best performing LightGBM model. In terms of precision, the model outperformed the second-best model by 3.4%, demonstrating better performance in predicting positive instances. For recall, our model showed an improvement of 8.9% compared to the second-best model, indicating its stronger ability to identify positive instances. F1 score, as a comprehensive metric considering precision and recall, also exhibited a 6.3% improvement in our model, which comprehensively reflects the superiority of the model.

Furthermore, to visually demonstrate the superior classification ability of our model, we graphically compared the performance of the GWO-LSTM-HNA model with traditional classification models using ROC curves. Fig. 7(left):

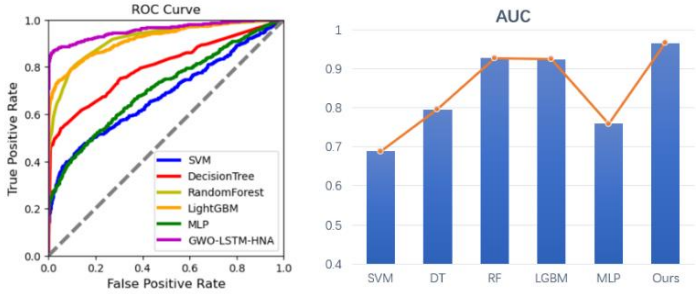

Fig. 7. left: ROC curve for traditional methods. right: Comparison of AUC for traditional methods.

It illustrates the ROC curves of the traditional classification methods, clearly indicating the significant advantages of our proposed model in classification prediction capability. Additionally, we utilized AUC values to plot Fig.7(right), providing a more intuitive display of the classification abilities of each model.

To further validate the effectiveness of our proposed GWO-LSTM-HNA network model, this study conducted a comparative analysis with relevant research results published in the field of power equipment defect prediction in recent years. Specifically, we compared our model with the SMOTE-XGBoost method [6] and the LSSVM-BNDT [10] method. The former was proposed by Wang et al. in 2021, which is based on SMOTE-XGBoost for transformer defect prediction, while the latter was proposed by Jia et al. in the same year, based on least squares support vector machine Bayesian network decision tree for defect diagnosis.

Table 3

Comparison experiments with relevant methods

Method	Precision	Recall	F1-score	Accuracy
SMOTE-XGBoost	94.08	88.52	83.91	93.08
LSSVM-BNDT	94.77	84.07	80.09	91.96
GWO-LSTM-HNA	**97.39**	**91.65**	**94.38**	**97.37**

In the experiment, we recorded the performance of these methods and conducted a comprehensive evaluation using consistent evaluation metrics. As shown in Table , our model achieved an improvement of 4.29% and 5.41% in accuracy compared to the SMOTE-XGBoost and LSSVM-BNDT methods, respectively. In other key performance metrics precision, recall, and F1 score our model also demonstrated significant advantages. Particularly, in terms of the F1 score, compared to the SMOTE-XGBoost method, we achieved a 10.47% improvement, highlighting the significant enhancement of our model in overall performance.

Furthermore, to visually demonstrate the classification prediction capability of our model, we also plotted ROC curves and utilized AUC values for quantification (Fig. 8), which clearly showcase the superiority of our method.

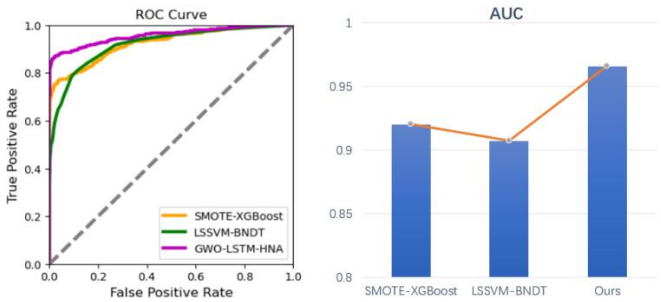

Fig. 8. left: ROC curve for relevant method. right: Comparison of AUC for relevant methods.

3.4 Ablation Experiments

To validate the superiority of our proposed GWO-LSTM-HNA model and identify the key contributions of each component in the model, this study conducted further ablation experiments based on comparative experiments. By gradually removing the key components of the model, we evaluated the impact of these components on the overall performance of the model. Specifically, experiments were conducted to remove the Grey Wolf Optimization (GWO) module and the Hidden Layer Neuron Attention (HNA) module.

Table 4

Ablation experiment

Method	Precision	Recall	F1-score	Accuracy
LSTM	96.36	86.53	90.62	95.80
GWO-LSTM	96.37	87.18	91.46	96.26
LSTM-HNA	95.92	91.20	93.37	96.86
GWO-LSTM-HNA	**97.39**	**91.65**	**94.38**	**97.37**

The experimental results summarized in Table show the changes in various metrics. The results indicate that the GWO module, by optimizing hyperparameters adjustments, improved the accuracy of the model by 0.5%. Furthermore, the HNA module, by addressing the issue of redundant features, further increased the model's accuracy by 1.1%. These findings confirm the effectiveness of both modules, with the neuron attention mechanism module making a significant contribution to performance improvement. When these two modules were combined with the LSTM network, the model achieved the highest accuracy, improving by 1.6% compared to the original LSTM network. To visually demonstrate the results of the ablation experiments and to substantiate the superiority of the proposed method, we plotted the ROC curves and AUC comparison chart for the ablation experiments (Fig. 9).

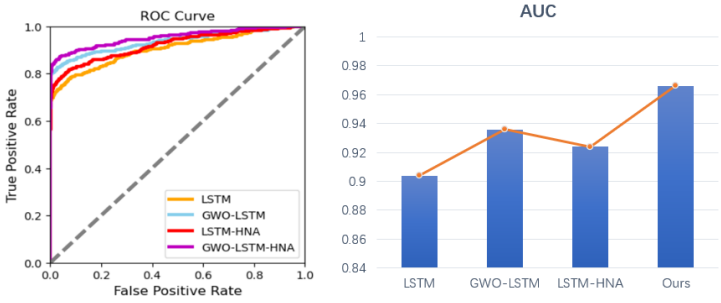

Fig. 9. left: ROC curve for ablation experiment. right: Comparison of AUC for ablation experiment.

From the chart, it is evident that the performance of the LSTM model significantly improved after integrating the GWO and HNA modules, with the GWO-LSTM-HNA model exhibiting the most outstanding performance, surpassing the other three models.

4. Conclusion

Defect prediction in power equipment aims to utilize historical defect data to forecast potential issues in the future. Although existing prediction methods, such as transformer defect prediction based on SMOTE-XGBoost and LSTM-based transformer fault prediction, have achieved some success, they still have shortcomings. Firstly, these methods fail to fully consider the influence of temporal relationships on predictions. Secondly, even when considering temporal relationships, they do not effectively handle features, leading to an inability to reduce interference from non-key features, thus impacting the efficiency and prediction accuracy of the models. Additionally, the adjustment of hyperparameters plays a crucial role in determining model performance, but existing methods do not provide effective tuning solutions.

In view of these issues, this study proposes a transformer defect prediction method that combines the Grey Wolf Optimizer (GWO) with LSTM-HNA. This method adopts LSTM networks to learn temporal features from historical data and innovatively designs a hidden layer neural attention mechanism (HNA) to enable the model to focus more on important features. Simultaneously, it automatically searches for the optimal combination of hyperparameters using the Grey Wolf Optimizer algorithm. Ultimately, we use a joint training strategy to train the model and validate the effectiveness and superiority of the proposed method through comparative experiments and ablation studies.

Funding Statement: The authors gratefully acknowledge the support of the "Anhui Province Science and Technology Major Special Project (202203a05020023)".

R E F E R E N C E S

[1]. *Yao Zhang, Aohan Wang, Hong Zhang.* Overview of the Development of China's Smart Grid[J]. Power System Protection and Control, 2021,49(05): 180-187.

[2]. *Xuehui Ding, Hailin Xv, Yingting Luo, et al*. Transformer DGA fault diagnosis based on random forest feature selection and MAEPSO-ELM algorithm[J]. Journal of Electric Power Science and Technology, 2022,37(02):181-187.

[3]. *Huidong Wang, Haiyan Yao, Qiang Guo, et al*. Transformer Fault Diagnosis Method Based on Multi-Scale Convolutional Neural Network[J]. Journal of Electric Power Science and Technology, 2023,38(04):104-112.

[4]. *Zhenfei Chen, Huangyong Zhang, Hongzhong Ma, et al*. Review of Fault Diagnosis Methods for Power Equipment Based on Deep Learning[J]. Electrical Automation, 2022,44(01):1-2.

[5]. *Xiaofan Zhao, Shuming Du.* Research on Fault Diagnosis Methods for Substation Equipment Based on Power Big Data[J]. Information Technology, 2022,46(09):163-168.

[6]. *Wenbo Wang, Xiaomei Zeng, Yinchuan Zhao, et al*. Transformer Defect Prediction Based on SMOTE-XGBoost[J]. Journal of North China Electric Power University, 2021,48(05):54-60.

[7]. *Xiaohu Zhang, Huanyu Ning.* " Transformer fault diagnosis method based on ReliefF and HPO-SVM". U.P.B. Sci. Bull., Series C, Vol. 85, Iss. 3, 2023

[8]. *Yuping Yan, Zhanhui Xiao.* Fault Prediction Algorithm for Power Grid Based on Autoencoder Neural Network[J]. Journal of Shenyang University of Technology, 2023,45(01): 1-5.

[9]. *Like Dong, Lu Bai, Na Wu, et al*. Research on Power Transformer Fault Prediction Method Based on Knowledge Graph[J]. High Voltage Electrical Equipment, 2022,58(11):151-159.

[10]. *Y. Jia, L. Ying, D. Wang, et al*. Defect Prediction of Relay Protection Systems Based on LSSVM-BNDT[J]. in IEEE Transactions on Industrial Informatics, vol. 17, no. 1, pp. 710-719, Jan. 2021.

[11]. *Z. Chen, L. Hong, Y. Gu, et al*. A novel support vector machine multi-classification strategy for power transformer fault diagnosis[C]. 2022 IEEE 5th International Electrical and Energy Conference (CIEEC), Nangjing, China, 2022, pp. 3368-3373.

[12]. *Lei Wang, Changzheng Chen*. Research on Fault Prediction of Transformers[J]. Mechanical Design and Manufacturing, 2023(01):65-68

[13]. *H. Huang, Y. Lv, L. Xue, et al*. Power Grid Fault Diagnosis based on Support Vector Machine[C]. 2021 IEEE 5th Conference on Energy Internet and Energy System Integration (EI2), Taiyuan, China, 2021, pp. 2301-2307.

[14]. *Ivan S, Evgeny L, Andrey R, et al*. Power Equipment Defects Prediction Based on the Joint Solution of Classification and Regression Problems Using Machine Learning Methods[J]. Electronics,2021,10(24).

[15]. *Yong Yu, Xiaosheng Si, Changhua Hu, et al.* A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures[J]. Neural Compute, 2019,31(7):1235–1270.

[16]. *Dongmei Chen*. Comprehensive Review of the Current Research on LSTM[J]. Information Systems Engineering, 2022(01):149-152.

[17]. *Sherstinsky A.* Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network[J]. Physica D: Nonlinear Phenomena, 2020, 404: 132306.

[18]. *Zhangli Zhu, Yuan Rao, Yuan Wu, et al*. Research Advances in Attention Mechanism in Deep Learning[J]. Chinese Journal of Information Science, 2019,33(06):1-11.

[19]. *Huan Ren, Xvguang Wang*. Comprehensive Review of Attention Mechanism[J]. Computer Applications, 2021,41(S1):1-6.

[20]. *Z. Niu, G. Zhong, H. Yu*. A review on the attention mechanism of deep learning[J]. Neurocomputing, 2021, 452: 48-62.

[21]. *Jiayuan Li, Yaonan Li, Jilu Hui*. Comprehensive Review on the Application of Grey Wolf Optimization Algorithm[J]. Digital Technology and Applications, 2022,40(09):10-13.

[22]. *Mirjalili S, Mirjalili S M, Lewis A*. Grey Wolf Optimizer[J]. Advances in Engineering Software, 2014, 69(3):46–61.

[23]. *Xue Xing, Yaqi Zhai.* "SSA-Conv-LSTM-based time-space demand forecasting method for online car reservation". U.P.B. Sci. Bull., Series C, Vol. 86, Iss. 1, 2024

[24]. *Xiao-Hong Yin, Ran Song, Zhi-Ding Chen et al*. "PSO-LSTM based construction schedule prediction method for shield tunneling". U.P.B. Sci. Bull., Series C, Vol. 86, Iss. 1, 2024