Heliyon 9 (2023) e16938

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Ultra-short-term wind power prediction method combining financial technology feature engineering and XGBoost algorithm

Shijie Guan [a,b], Yongsheng Wang [a,b,*], Limin Liu [a,b], Jing Gao [c], Zhiwei Xu [a,b], Sijia Kan [d]

[a] School of Data Science and Application, Inner Mongolia University of Technology, Hohhot 010080, China
[b] Software Service Engineering Technology Research Center, Inner Mongolia Autonomous Region, Hohhot 010080, China
[c] School of Computer and Information, Inner Mongolia Agricultural University, Hohhot 010018, China
[d] School of Natural Sciences, The University of Manchester, Manchester, M13 9PL, UK

ARTICLE INFO

Keywords:
Industrial applications of financial technical indicators
Gradient boosting regression trees
Parameter optimization theory
Wind power prediction

ABSTRACT

The input features of existing wind power time-series data prediction models are difficult to indicate the potential relationships between data, and the prediction methods are based on deep learning, which makes the convergence of the models slow and difficult to be applied to the actual production environment. To solve the above problems, an ultra-short-term wind power prediction model based on the XGBoost algorithm combined with financial technical index feature engineering and variational ant colony algorithm is proposed. The model innovatively applies financial technical indicators from financial time series data to wind power time series data and creates a class of model input features that can highly condense the potential relationships between time series data. A bionic algorithm is used to search for the best computational parameters for financial technical indicators to reduce the reliance on financial experts' experience. Taking the German power company Tennet wind power data set as an example, the prediction model proposed in this study has an mean absolute error of 0.859 and a root mean square error of 1.329, and it takes only 244 ms to complete the prediction. Thus, this study provides a new solution for ultra-short-term wind power prediction.

1. Introduction and related work

In light of the rapid development of the global economy in the 21st century, energy demand has surged, while traditional fossil fuel

Abbreviations: CART, Classification and Regression Tree; CNN, Convolutional Neural Network; ENTSO-E, European Network of Transmission System Operators for Electricity; FTI, Financial Technical Indicator; FTIs, Financial Technical Indicators; GAN, Generative Adversarial Network; GRA, Grey Relational Analysis; GRU, Gated Recurrent Unit Network Algorithm; GW, Gigawatt; IHBA-SVM, Improved Honey Badger Algorithm-Support Vector Machine; KDJ, Stochastic Oscillator; LightGBM, LightGBM Algorithm; LSTM, Long and Short-Term Memory Network Algorithm; MACD, Moving Average Convergence and Divergence; MAE, Mean Absolute Error; NWP, Numerical Weather Prediction; PCA, Principal Component Analysis; PG-GAN, Progressive Growing of Generative Adversarial Network; RMSE, Root Mean Square Error; RSI, Relative Strength Index; SARIMA, Seasonal Autoregressive Sliding Average; Seq2Seq, Sequence to Sequence; SVM, Support Vector Machine; TCN, Temporal Convolutional Network Algorithm; XGBoost, eXtreme Gradient Boosting Algorithm.

* Corresponding author. School of Data Science and Application, Inner Mongolia University of Technology, Hohhot 010080, China.
E-mail addresses: 20211800688@imut.edu.cn (S. Guan), wangys@imut.edu.cn (Y. Wang), liulimin789@126.com (L. Liu), gaojing@imau.edu.cn (J. Gao), xuzhiwei2001@ict.ac.cn (Z. Xu), sijia.kan@postgrad.manchester.ac.uk (S. Kan).

https://doi.org/10.1016/j.heliyon.2023.e16938
Received 2 April 2023; Received in revised form 30 May 2023; Accepted 1 June 2023
Available online 2 June 2023

sources face the threat of depletion. Renewable energy has emerged as a viable solution to address energy security and the climate crisis [1]. The wind power industry experienced a remarkable year in 2022, with a staggering 78 GW of new installed capacity worldwide. This represents the third-highest annual increase in history, and total installed capacity is anticipated to rise to 906 GW, reflecting an impressive 9% year-on-year increase. Wind power has gained significant momentum as a practical response to energy security and the climate crisis. Precise wind power generation prediction is vital to the development of effective power consumption plans, implementation of power generation tasks, and adjustment of national strategies. Additionally, accurate wind power prediction is an essential step towards achieving net-zero emissions and carbon neutrality [2].

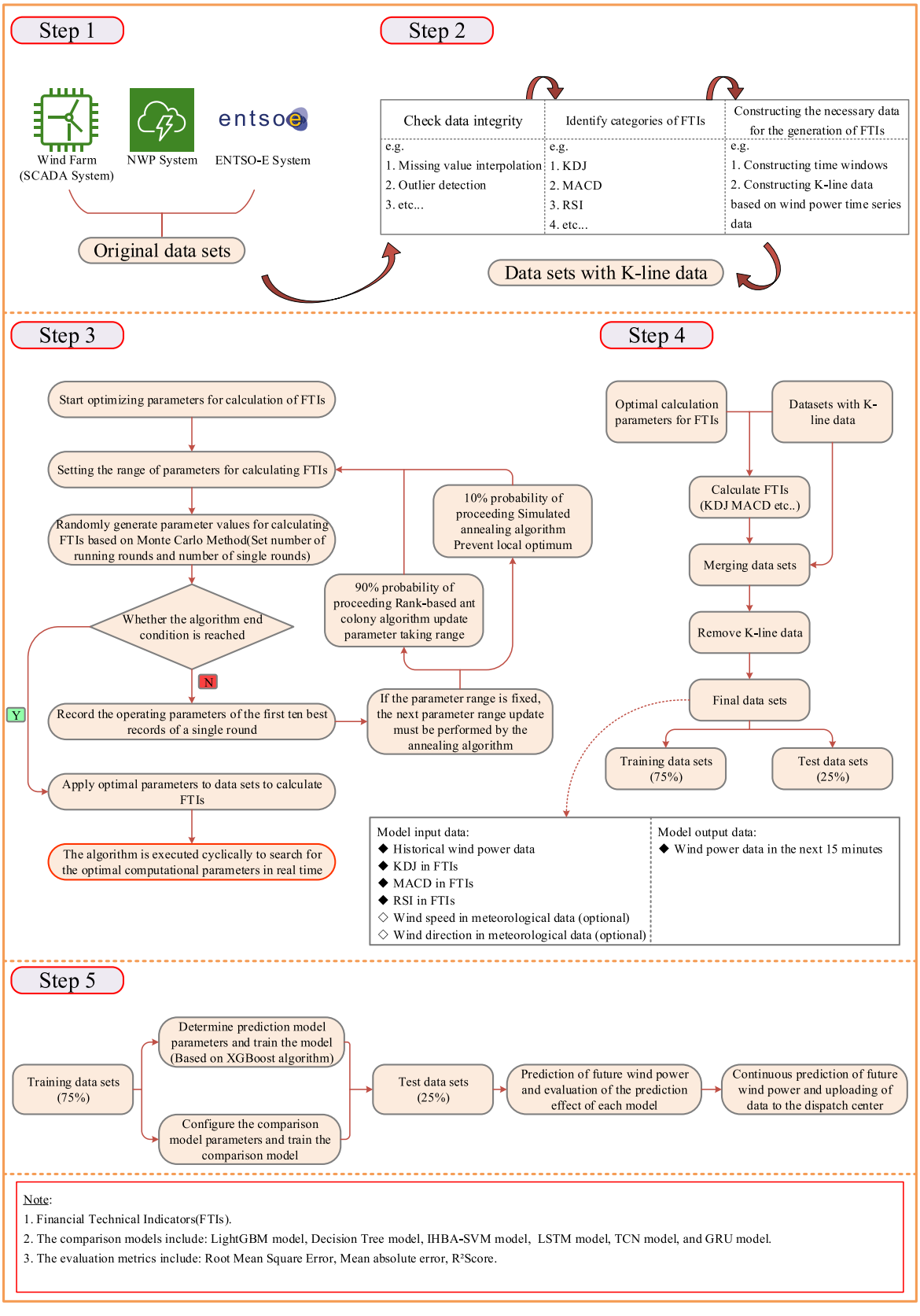
However, the uncertainty of natural conditions such as wind speed and direction makes wind power generation highly random, volatile, and uncontrollable. This will lead to a serious wind abandonment phenomenon, and the grid connection of wind power will be somewhat affected. In addition, the accurate prediction of ultra-short-term wind power becomes challenging. Researchers have designed many ultra-short-term wind power prediction models to address this problem, including statistical regression models, machine learning models (including deep learning models), and combined models using parametric optimization methods.

Statistical models are autoregressive models, moving average models, and correlated variant models. They are used in small sample, linear and smooth wind power prediction tasks because of their ease of computation and flexibility [3]. For example, Zhang W et al. used the SARIMA model to effectively extract signals with seasonal characteristics from wind power time series data considering the influence of season on wind power time series data, which improved the prediction accuracy of the model. However, the missing data may cause the model to deviate from the actual application scenario [4]. Zhang F et al. used a dynamic adaptive prediction model based on an autoregressive model. The model coefficients were adaptively updated according to wind power time series data characteristics, which improved the model's prediction accuracy. However, abnormal data can significantly reduce the prediction accuracy of this model and make the prediction results lag [5]. In summary, statistical models only analyze the potential relationships between time series data and can hardly be used to explore their nonlinear relationships. Therefore, such models are only applicable to static data analysis, which is an obvious disadvantage of such models. Wind power time series data are usually large sample data with nonlinear characteristics. There are often limitations when applying such models to wind power prediction [6].

With the rapid development of artificial intelligence, machine learning algorithms and deep learning algorithms have been widely used in wind power prediction tasks [7]. Shi K et al. improved the random forest algorithm and removed the redundant features using the variable importance measure method, improving the model prediction accuracy. However, there is a defect of poor model interpretability [8]. Ju Y et al. improved the accuracy and robustness of the prediction model by combining the LightGBM algorithm with the CNN algorithm. However, the model will occupy more computational resources, with room for further performance improvement [9]. Sasser C et al. constructed a wind power prediction model combining physical features and a decision tree algorithm, and their use of feature engineering improved the model's prediction accuracy. However, the model requires extremely high input features [10]. Dong J et al. combined the XGBoost algorithm with a weather prediction correction system, which effectively improved the model defects caused by data errors but led to the model's prediction accuracy being closely related to the season [11]. Machine learning-based prediction methods can learn their features adaptively based on the data and have higher prediction accuracy than statistical models. However, these models have high requirements for data variety and completeness.

Today, deep learning is evolving rapidly. This technique has a wide range of applications in various fields. Among the common deep learning algorithms, Farah S et al. combined the LSTM algorithm with the GRU algorithm to achieve multi-step prediction of ultra-short-term wind power [12]. Zhang Y et al. used a deep learning method with an attention mechanism combined with the Seq2Seq model to improve the model prediction accuracy. However, the large amount of parameter setting work significantly affected the performance of this model [13]. Chengqing Y et al. combined graph attention network, GRU algorithm, and TCN algorithm to effectively extract features from wind power time series data, which significantly improved the accuracy and robustness of the model, but the model needed to define the graph structure in advance and could not explore the deep relationships between data [14]. Yuan R et al. improved the GAN algorithm. They proposed a PG-GAN-based wind power prediction algorithm, which can improve efficiency while ensuring model prediction accuracy, but the model does not consider meteorological features [15]. In summary, deep learning models can better exploit inter-data information and effectively improve model prediction accuracy compared with statistical and machine learning models. However, the increase in data volume, especially when the model is too complex, requires more computational resources and training time. It is not easy to meet the task timeliness of ultra-short-term wind power prediction. Meanwhile, the interpretability of the deep learning model still has room for further improvement.

However, machine learning and deep learning models require setting many parameters, and the hyperparameters set by expert experience often differ from the optimal parameters required by the model. Today, the growing popularity of bionic algorithms is bringing new ideas to solve real-world problems. The basic rules followed by many tiny creatures in nature create collective intelligence at the macro level. The way of life of swarming insects perfectly illustrates this mode of existence, which is very different from that of human society. Therefore, many combinatorial models for ultra-short-term wind power prediction have emerged with the rapid development of parameter optimization methods based on bionic algorithms. Li L et al. constructed a wind power prediction model based on a wavelet decomposition algorithm, support vector machine algorithm, and atomic optimization algorithm, which optimized the hyperparameters of the support vector machine and reduced the time required for prediction but was too demanding on the input features [16]. Liu M et al. constructed a prediction model based on the SVM and grey wolf optimization algorithms, optimizing the model's hyperparameters. However, its application to stock screening strategies ignores the allocation of unequally weighted capital [17]. Li C et al. constructed a prediction model based on the SVM and cuckoo search algorithm. This model can effectively handle the errors while the support vector machine hyperparameters are further optimized. However, this model cannot be applied to long-term wind power prediction or identify abnormal data [18]. In summary, current parametric optimization methods based on bionic algorithms are often applied to the hyperparameter optimization of models. Generally, the optimized models have better prediction



Note:
1. Financial Technical Indicators(FTIs).
2. The comparison models include: LightGBM model, Decision Tree model, IHBA-SVM model, LSTM model, TCN model, and GRU model.
3. The evaluation metrics include: Root Mean Square Error, Mean absolute error, R^2Score.

(caption on next page)

Fig. 1. Model construction and experimental flow chart.

performance but require longer prediction times.

The authors found in many wind power prediction studies that the input features of the model are mostly historical and meteorological data. The single features make it difficult to grasp the potential relationships between the data, and the redundant meteorological data increase the model's computational burden and reduce the model's prediction speed. Meanwhile, acquiring highly accurate meteorological data imposes a certain economic burden on the electric field. In addition, most ultra-short-term wind power prediction models are based on deep learning models, which take much time for training and prediction. Ultra-short-term wind power prediction task often predicts wind power after 15 min. However, the model training and prediction time are more than 15 min, so such models cannot be practically applied to wind farms. Research has shown that machine learning models combined with well-designed feature engineering can yield prediction accuracies that rival or exceed deep learning models [19].

The main contributions of this study are twofold :

- An input feature for an ultra-short-term wind power prediction model is provided. By looking at the financial time series and the wind power time series, the authors argue that there is some similarity and that the wind power time series is likely to have a similar pattern or nature to the financial time series [6]. FTIs are common and important model input features in financial time series data prediction. Using FTIs to predict financial market movements is a common method [20]. In this study, K-line and FTIs are constructed using wind power time series data, and this project is named financial technology feature engineering. Experiments show that FTIs based on wind power time series data can significantly improve model prediction accuracy when used as model input features, replacing meteorological data in common model input features.
- A parameter optimization method applicable to the present model when performing feature calculations is provided. FTIs of wind power time-series data should not rely on financial experts' experience in their calculation and need to be calculated using parameters that match the current data. This study transforms this problem into a multivariate function optimal solution problem, and a variational ant colony algorithm is designed to search for optimal parameters for calculating FTIs, using the bionic algorithm as an entry point [21]. The algorithm is a combination of the Monte Carlo algorithm [22], rank-based ant colony algorithm [23], and simulated annealing algorithm [24]. Experiments show that it effectively solves the over-reliance on financial experts' experience when calculating FTIs.

This study uses a wind farm data set in Inner Mongolia, China, to verify the validity of the FTIs, a UK onshore wind data set to verify the effectiveness of the variational ant colony algorithm for optimizing the parameters of the FTIs calculation, a German power company Tennet wind power data set to verify the superiority of the prediction model proposed in this study.

The rest of the paper is organized as follows: Section 2 presents the computation of the FTIs in the model, the workflow of the variational ant colony algorithm, and the derivation process of the objective function of the XGBoost algorithm. Section 3 details the data sets used in the experiments and the evaluation metrics. Section 4 conducts ultra-short-term wind power prediction experiments and analyzes the results. Section 5 discusses the importance of FTIs and variational ant colony algorithms. Section 6 concludes the paper.

2. Key algorithms

2.1. Model introduction

There are five steps to construct the prediction model designed in this study, and the main work performed in each step is as follows. A schematic diagram of the model construction and the experimental process is shown in Fig. 1. The diagram was drawn concerning the drawing method of the schematic diagram in reference [25]. In addition, the presentation of parts of this paper mimics the presentation in Ref. [26].

- Step1: Collect the required raw data from the wind farm/ENTSO-E system/NWP system and construct the original data set.
- Step2: (1) Check data integrity. (2) Determine the categories of FTIs needed for the model. (3) Construct time window and K-line data to prepare for the calculation of FTIs.
- Step3: (1) Set the search range for the calculation parameters of FTIs. (2) Setting the number of rounds of algorithm operation. (3) Generate computational parameters of FTIs by the Monte Carlo method in the specified range. (4) Record the top 10 groups of parameters with the best results in each round of experiments, and set each parameter's maximum and minimum values in these 10 groups of parameters as the parameter search range in the next round of experiments. In addition, to avoid the problem of an optimal local solution, there is a certain probability that the search range will be expanded when reset. When the maximum value of the parameter search range is the same as the minimum value, the search range will be forced to expand. (5) When the number of rounds of algorithm operation is reached, the calculated parameters of FTIs corresponding to the best experimental results are output.
- Step4: (1) Calculate the FTIs based on the K-line data provided in step 2 and the optimal financial technical indicator calculation parameters provided in step 3. (2) Combine the FTIs into the data set and remove the K-line data. Use this data set to divide the model training set and test set.

- Step5: (1) Train the prediction models proposed in this study with the comparison models using the training set. (2) Use the test set to verify the prediction performance of all models. (3) Report the prediction results to the dispatch center to assist in generation planning and power consumption schedule.

2.2. FTIs construction algorithm

The calculation of FTIs is inseparable from the data in financial K-line charts. Before calculating the FTIs based on wind power time series data, a suitable time window should be selected and converted to K-line data. Then, the FTIs should be calculated according to the relevant algorithm.

Wind power timing data are generally stream data collected 24 h a day without interruption. This study aims to solve the ultra-short-term wind power prediction problem (predicting the wind power in the next 15 min), so the time window is set to 1 h to calculate the FTIs. It further means that the data acquisition step of this study is 5, using 5 points of data to synthesize 1 K-line data. Shorter acquisition steps would lead to distortion of the FTIs and would not effectively capture the characteristics and trends in the wind power time series data. Longer acquisition steps would lead to blunting of the FTIs, causing them to lag and not synchronize with the current trend.

A rectangular bar in the daily K-line of the financial markets represents a trading day's open, close, high, and low prices. If the opening price is lower than the closing price, the bar is shown in red (generally in China, but in the U.S. market, it is in green). If the opening price is higher than the closing price, the bars are shown in green (generally in China, but in the U.S. market, it is in red). A detailed explanation of the K-line is shown in Fig. 2.

The open, close, high, and low prices in the K-line data can be replaced by the start, end, high and low values in the time window of the wind power time series data. In this study, the time window is set to 1 h to calculate the K-line data based on the above contents. The calculation process is shown in Fig. 3. Taking the red dashed line as an example, in steps 1–2 of Fig. 3, this study combined the 5-time series data corresponding to 00:00–01:00 into 1 K-line data. Further, the K-line graph was plotted in step 3 of Fig. 3 based on the K-line interpretation of Fig. 2, where the horizontal axis represents the time collection point, and the vertical axis represents the power value.

2.2.1. Financial technical indicators KDJ

KDJ first originated in the futures market and was pioneered by George C. Lane. It combines the advantages of the momentum concept, strength and weakness indicators, and moving averages. It measures the degree of variation of values from the normal interval [27]. In the calculation of KDJ data, there are variables in its formula that are used to limit the calculation period and weights. For the ultra-short-term wind power prediction, the calculation period is set to 3 h. The details of the algorithm are shown below.

Algorithm input: The wind power time series data. Where the power value at the end of 3 h is recorded as C_3, the highest power value during 3 h is recorded as H_3, and the lowest power value during 3 h is recorded as L_3.

Algorithm steps:

a) RSV is the immature random value. Set the RSV value at hour x to RSV_x. K is the fast confirmation indicator. Set the K value at hour x to K_x. D is the slow confirmation indicator. Set the D value at hour x to D_x. j is the trend direction indicator, set the J value at hour x to J_x.

b) Based on $RSV = \frac{C_3 - L_3}{H_3 - L_3} \times 100$, calculate the value of RSV.

c) Let K_{x-3} be the K value for the first 3 h. If not present, use 50 instead. Using the RSV found in step b, calculate the K value according to $K_x = \frac{3}{4}K_{x-3} + \frac{1}{4}RSV_x$. The K value varies in the interval 0–100.

d) Let D_{x-3} be the D value for the first 3 h. If not present, use 50 instead. Using the K found in step c, calculate the D value according to $D_x = \frac{3}{4}D_{x-3} + \frac{1}{4}K_x$. The D value varies in the interval 0–100.

e) Using K from step c and D from step d, calculate the value of J from $J_x = 3K_x - 2D_x$.

Algorithm output: The K, D, and J values of the FTIs.

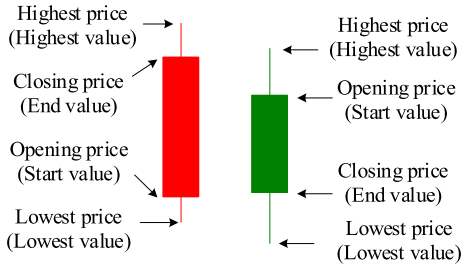


Fig. 2. Standard K-line detail chart.

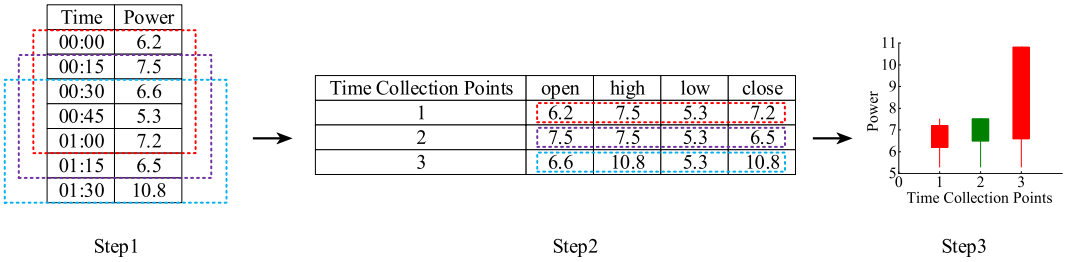


Fig. 3. Wind power K-line construction schematic.

2.2.2. Financial technical indicators MACD

MACD is a common technical analysis indicator in financial market trading, introduced by Gerald Appel in the 1870s. It is used to study the strength, direction, and energy of stock price changes and the trend cycle [28]. In this study, MACD is migrated from financial time series data to wind power time series data to determine the strength, direction, trend, etc., of forecast target changes. For ultra-short-term wind power forecasting, the fast-moving average calculation period in MACD is set to 3 h, and the slow-moving average calculation period is set to 5 h in this study. The details of the algorithm are shown below.

Algorithm input: The wind power time series data. where the power value at the end of x hours is recorded as C_x.

Algorithm steps:

a) Let EMA_x^{fast} be the average of the fast-moving line at hour x. EMA_{x-3}^{fast} is the EMA^{fast} value for the first 3 h with hour x as the base. If the value does not exist, use 0 instead. According to $EMA_x^{fast} = \frac{2}{3}EMA_{x-3}^{fast} + \frac{1}{3}C_x$, the EMA^{fast} value is calculated, and the EMA^{fast} value can be negative.

b) Let EMA_x^{slow} be the average of the slow-moving line at hour x. EMA_{x-5}^{slow} is the EMA^{slow} value for the first 5 h with hour x as the base. If the value does not exist, use 0 instead. According to $EMA_x^{slow} = \frac{4}{5}EMA_{x-5}^{slow} + \frac{1}{5}C_x$, the EMA^{slow} value is calculated, and the EMA^{slow} value can be negative.

c) Let DIF_x be the deviation value at hour x. Using the EMA^{fast} found in step a, and the EMA^{slow} found in step b, calculate the DIF_x value according to $DIF_x = EMA_x^{fast} - EMA_x^{slow}$.

d) Let DEA_x be the smoothed moving average at hour x. Using the EMA^{fast} found in step a with the EMA^{slow} and EMA_{x-5}^{slow} found in step b, calculate the DEA_x value according to $DEA_x = EMA_x^{fast} + \frac{3}{4}EMA_{x-5}^{slow} + \frac{1}{4}EMA_x^{slow}$.

e) Let $MACD_x^{bar}$ be the MACD value at hour x. Use the EMA^{fast} found in step a with the EMA^{slow} and EMA_{x-5}^{slow} found in step b to calculate the $MACD_x^{bar}$ value based on $MACD_x^{bar} = EMA_x^{fast} + \frac{3}{4}EMA_{x-5}^{slow} - \frac{7}{4}EMA_x^{slow}$.

Algorithm output: The DIF, DEA, and MACD values of the FTIs.

2.2.3. Financial technical indicators RSI

RSI is a technical analysis indicator proposed by Willis Wilde, an American mechanical engineer, in June 1978. In this study, the RSI indicator from financial time series data is migrated to wind power time series data. All three financial technical indicators, MACD, KDJ, and RSI, are used as model input features to construct an ultra-short-term wind power prediction model. The details of the algorithm for the construction of the financial technical indicator RSI are shown below.

Algorithm input: The wind power time series data. where the power value at the end of x hours is recorded as C_x.

Algorithm steps:

a) Let the magnitude of the upward fluctuation of power in 6 h be A, and let the magnitude of the downward fluctuation of power in 6 h be B.

b) If $C_x - C_{x-1} > 0$, then $A+ = C_x - C_{x-1}$. If $C_x - C_{x-1} < 0$, then $B+ = C_{x-1} - C_x$. Calculate the values for 6 consecutive sampling points.

c) $RS = \frac{A}{B}$, $RSI1X = \left(1 - \frac{1}{1+RS}\right) \times 100\%$.

d) Usually, the RSI1X, RSI2X, and RSI3X need to be calculated. RSI1X represents the value of 6 consecutive samples, RSI2X represents the value of 12 consecutive samples, and RSI3X represents the value of 24 consecutive samples.

Algorithm output: The RSI1X, RSI2X, and RSI3X values of the FTIs.

Meanwhile, this study aims to apply FTIs to wind power time series data. Since there are some differences between financial time series data and wind power time series data, the financial experts' experience (e.g., the financial technical indicator KDJ has a calculation period of 3 h) cannot be directly applied to the calculation of FTIs for wind power time series data. Therefore, this study proposes a variational ant colony algorithm to solve this problem.

2.3. Variational ant colony algorithm

FTIs in financial time series data require a large number of parameters to be set in the calculation, and the existing parameter setting schemes are mostly based on financial experts' experience. This study proposes a parameter adaptive optimization algorithm based on a variational ant colony algorithm to make the FTIs better fit the required predicted wind power time series data. The algorithm combines the Monte Carlo method, simulated annealing algorithm, and rank-based ant colony algorithm. The idea is to set the parameters randomly by the Monte Carlo method in the limited parameter search range, number of running rounds, and times. The rank-based ant colony algorithm narrows the parameter search range. A simulated annealing algorithm prevents the optimal local solution. Meanwhile, due to the extremely fast computation speed of the XGBoost algorithm, the variational ant colony algorithm can dynamically adjust the computation parameters of the FTIs according to the recent time series data to achieve the best fitting effect. The flowchart of the algorithm operation is shown Fig. 4.

2.3.1. Monte Carlo method

Statistically, the three types of FTIs used in this study require the setting of six parameters. More than 600,000 parameter setting options exist after delineating the parameter search range based on the short-term prediction objectives. According to the measurement, the XGBoost algorithm used in this study takes about 250 ms to complete a prediction (the average prediction time for all data sets). After traversing all setting schemes, the program runs for more than 40 h, which significantly reduces the usability of this model.

To this end, this study uses the Monte Carlo method to solve the parameter setting problem of FTIs so that the parameter setting of FTIs no longer depends on expert experience. It also benefits from the efficient and parallel computation of the Monte Carlo method, which significantly reduces the optimal parameter search time.

2.3.2. Rank-based ant colony algorithm

Setting parameters for calculating FTIs no longer relies on expert experience. However, there are still problems with the inefficient search for optimal parameters and large search ranges. For this reason, this study designs a rank-based ant colony algorithm to solve this series of problems.

The development of ant colony algorithms is closely related to the foraging habits of ants. The pheromones left on the ground by ants can guide the colony to find food effectively [29]. Further, this study used a rank-based ant colony algorithm in which the top 10 records in each round of R2Score ranking were considered the best ants. The records corresponded to parameters similar to pheromones in the ant colony algorithm. Before the next round of parameter search, the search range of each parameter is updated to the maximum and minimum values of this parameter in these 10 records. The rank setting in the rank-based ant colony algorithm is

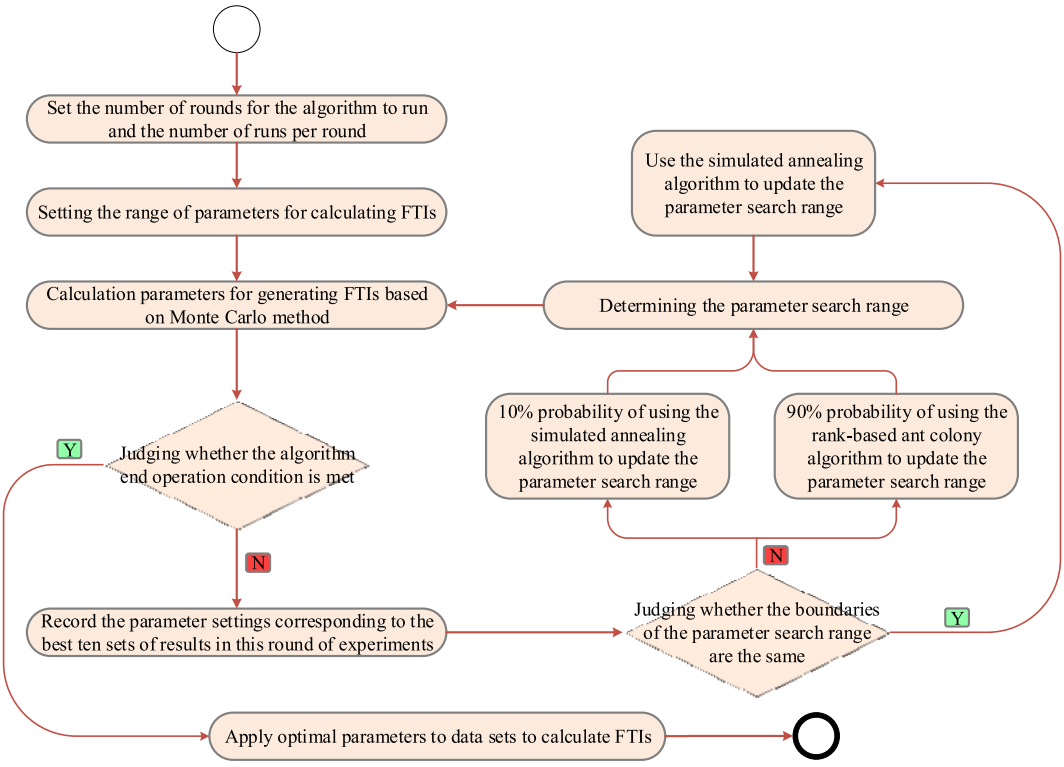


Fig. 4. Variational ant colony algorithm running flowchart.

described in detail in Section 4.1.2.

In addition, the experimental results show that after adding the rank-based ant colony algorithm, the parameter search range is rapidly reduced, the parameter search efficiency is significantly improved, and the model usability is substantially enhanced.

2.3.3. Simulated annealing algorithm

According to the method described in Section 2.3.2, the search efficiency of the optimal parameters has been significantly improved. However, after a certain number of rounds of operations, a situation arises where the upper bound of the search range of certain parameters to be determined is equal to the lower bound. At this time, the model has a problem with local optimal solutions. This study adds the simulated annealing algorithm to the variational ant colony algorithm to solve the problem of locally optimal solutions.

In this study, two simulated annealing schemes are proposed. One option is probabilistic annealing, which has a 10% probability of doubling the search range of all parameters before each round of parameter search range update (the setting of this probability will be described in detail in Section 4.1.3), and this is done to reduce the cumulative impact of the rank-based ant colony algorithm on the optimal local solution. Another option is forced annealing, which necessarily doubles the search range of all parameters to search for the optimal global solution when the maximum value of the partial parameter search range is the same as the minimum value. In addition, the flow of the simulated annealing algorithm is described in detail in Fig. 4.

At this point, the variational ant colony algorithm is constructed. Taking the optimization of the calculation parameters of the financial technical indicator MACD as an example, the parameter optimization strategy of the variational ant colony algorithm is shown below.

Algorithm input: The range of values of the parameters to be determined, the raw wind power time series data for which features need to be extracted.

Algorithm steps:

a) Let the search range for the fast-moving average period in the financial technical indicator MACD be $[Fast_{Min}, Fast_{Max}]$. Let the search range for the slow-moving average period in the financial technical indicator MACD be $[Slow_{Min}, Slow_{Max}]$. Let the search range for the weight value of DEA in the financial technical indicator MACD be $[Dea_{Min}, Dea_{Max}]$.
b) Set the number of running rounds, set the number of single rounds running.
c) Set the number of records used to update the parameter search range (set to 10 in this case), and set the probabilistic annealing trigger rate (set to 10% in this case).
d) Determine whether to perform probabilistic annealing or forced annealing.
e) Compare all recorded R^2Scores for this round. Select the parameter corresponding to the top 10 highest R^2Scores in the record and update the parameter search range. The updated search range is $[Fast_{Min}^{New}, Fast_{Max}^{New}]$ for the fast-moving average period, $[Slow_{Min}^{New}, Slow_{Max}^{New}]$ for the slow-moving average period, and $[Dea_{Min}^{New}, Dea_{Max}^{New}]$ for the DEA weight value.
f) Repeat steps b, c, d, and e until the experiment proceeds to the set number of running rounds. Compare the R^2Score recorded in the last round, select the best combination of parameters, and output the combination of parameters.

Algorithm output: The optimal calculation parameters of the FTIs MACD.

2.4. XGBoost algorithm

XGBoost comprises multiple CART trees and realizes the integrated learning of multiple CART trees through Gradient Tree Boosting. In the training phase, each decision tree learns the residual error between the target value and the sum of the predicted values of all previous trees. Similar To balanced binary trees, the decision tree construction process will select the optimal segmentation point according to the optimal characteristics. In the end, multiple decision trees make decisions together, and the results of all trees are accumulated as the final prediction result.

The unique feature of the XGBoost algorithm is that the objective function contains both a loss function and a regularization term. The loss function represents how well the model fits the data, and it calculates the second-order derivative of the loss function, which further considers the trend of the gradient change. The regularization term is used to control the complexity of the model. The regularization term of XGBoost has a penalty mechanism, and the higher the number of leaf nodes, the higher the penalty, thus limiting the number of leaf nodes. In addition, the biggest improvement of the XGBoost algorithm is the significant improvement in computational speed. The most time-consuming part of the tree model construction process is sorting the feature values to determine the best classification point. The XGBoost algorithm sorts the features before training the model. It dumps them into a block structure, which is reused during the computation to reduce the computational effort of the algorithm [30–32].

In summary and combined with references [33–35], the XGBoost model is defined as follows:

$$\widehat{y}_i^m = \sum_{z=1}^{Z} f_z(x_i) = \widehat{y}_i^{(m-1)} + f_t(x_m) \tag{1}$$

In Equation (1), $\widehat{y}_i^m$ is the prediction result of sample i after m iterations, Z is the set of all CART trees, f_z is the prediction result of the z-th tree, x_i is the i-th sample of the input, $\widehat{y}_i^{m-1}$ is the prediction result of the first m-1 trees, and $f_t(x_m)$ is the prediction result of the m-th

tree. Further, the XGBoost model objective function is as follows:

$$Obj = \sum_{i=1}^{n} l(y_i, \widehat{y}_i) + \sum_{j=1}^{r} \Omega(f_i) \tag{2}$$

In Equation (2), the objective function Obj consists of two terms; the first is the loss function, which is used to evaluate the error between the predicted and actual values of the model. Where l is the model loss function, n is the total number of data, i is the ordinal number of data, y_i is the i-th data actual value, and $\widehat{y}_i$ is the i-th data predicted value. The second term is the regularization term, which controls the model complexity and avoids overfitting. Where r is the total number of trees, j is the ordinal number of trees, f_j is the j-th tree, and Ω is the regularization function of the model. The Taylor series of the loss function is generally calculated to the second order, and the constant term is removed. When there are m CART trees, the objective function is as follows:

$$\begin{aligned} Obj &= \sum_{i=1}^{n} \left(g_i f_m(x_i) + \frac{1}{2} h_i f_m^2(x_i) \right) + \Omega(f_m) \\ g_i &= \partial_{\widehat{y}_i^{(t-1)}} l\left(y_i, \widehat{y}_i^{m-1}\right) \\ h_i &= \partial^2_{\widehat{y}_i^{(t-1)}} l\left(y_i, \widehat{y}_i^{m-1}\right) \end{aligned} \tag{3}$$

In Equation (3), g_i and h_i are the first-order and second-order derivatives. In addition, the regularization term in this objective function is defined as:

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^{T} w_j^2 \tag{4}$$

In Equation (4), w_j is the weight fraction of the j-th child node in tree f; T is the total number of leaf nodes in tree f; γ and λ are the custom parameters of XGBoost, γ is the penalty term of the L1 norm, γT controls the complexity of the tree through the number of leaf nodes and coefficients, thus suppressing the complexity of the model. λ is the penalty term of the L2 norm. $\frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$ is used to control the weight fraction of the leaf nodes to avoid overfitting.

3. Experimental preparation work

3.1. Data sets used in the experiments

3.1.1. Inner Mongolia, China wind power data set (IM WPD)

This data set records ultra-short-term wind power data from a wind farm in Inner Mongolia, China, from January 1, 2019, to May 22, 2019, with a collection frequency of 15 min/time. The data contained missing values, which were processed to make a total of 12,520 data. The prediction target is the wind power generation power value after 15 min. The data set comes with meteorological data, and this study plans to conduct ablation experiments on this data set. To demonstrate that the FTIs can effectively improve the model prediction accuracy and reduce the dependence of the model on NWP data.

3.1.2. UK-wide onshore wind power data set (UK WPD)

In this study, ultra-short-term wind power data for the UK from 2015 to 2020 were obtained by ENTSO-E with a collection frequency of 15 min/time. The data do not contain missing values and anomalies, totaling more than 200,000 data. The prediction target is the wind power generation power value after 15 min. This study plans to conduct an important validation experiment of variational ant colony algorithm on this data. To prove that the variational ant colony algorithm can effectively optimize the calculation parameters of FTIs and reduce the model's reliance on financial experts' experience.

3.1.3. German power company Tennet wind power data set (Tennet WPD)

In this study, ultra-short-term wind power data for the German power company Tennet from 2015 to 2020 were obtained by ENTSO-E with a collection frequency of 15 min/time. The data do not contain missing values and anomalies, totaling more than 200,000 data. The prediction target is the wind power generation power value after 15 min. This study plans to conduct model comparison experiments on this data set to obtain the prediction effects of different machine learning models and deep learning models to prove that the prediction model proposed in this study has an excellent performance in terms of prediction accuracy and feasibility.

Table 1
Key information of the data set.

Data set name	Collection time	Number of data	Train/Test ratio	Number of features	Predictive target
IM WPD	2019.1–2019.5	12520	75%/25%	14	Future wind power (15 min)
UK WPD	2015.1–2020.9	201500	75%/25%	10	Future wind power (15 min)
Tennet WPD	2015.1–2020.9	201500	75%/25%	10	Future wind power (15 min)

For readers to understand the wind power time series data used in this study more intuitively, the key information of the three data sets has been summarized in Table 1.

3.2. Evaluation metrics

3.2.1. Mean absolute error

MAE represents the average absolute error between the actual and predicted values in the test set, negatively correlated with the prediction effect. This indicator can accurately reflect the actual prediction error. It is worth noting that the same data set must be selected when using this indicator. It is not available to compare the model prediction effect between different data sets using this indicator.

Setting m as the total number of samples and n as the sample number. y_n and $\widehat{y}_n$ are the actual and predicted values of the n^{th} sample respectively. The calculation method of MAE is shown in Equation (5).

$$MAE = \frac{1}{m}\sum_{n=1}^{m}|y_n - \widehat{y}_n| \tag{5}$$

3.2.2. R^2Score

R^2Score is the most commonly used index to evaluate the degree of merit of the regression model. Setting $\bar{y}$ as the sample mean. The meanings of other variables are the same as those in MAE. If R^2score<0, then the prediction error is greater than the error of the mean, which means the current model is meaningless. If R^2score = 0, the predicted value equals the mean value, which means the model is still meaningless. If R^2score = 1, the predicted value equals the actual value, which means the model can make error-free predictions. Therefore, the closer the R^2score is to 1, the better the model prediction is. The calculation method of R^2Score is shown in Equation (6).

$$R^2 Score = 1 - \frac{\sum_{n=1}^{m}(y_n - \widehat{y}_n)^2}{\sum_{n=1}^{m}(y_n - \bar{y})^2} \tag{6}$$

3.2.3. Root mean square error

RMSE refers to the square root of the mean squared error between the predicted value of the model and the actual value. Because the dimensionality can be reduced, RMSE is easier to calculate and compare. The meanings of the variables in its calculation method are consistent with those in the MAE calculation method. The calculation method of RMSE is shown in Equation (7).

$$\mathrm{RMSE} = \sqrt{\frac{1}{m}\sum_{n=1}^{m}(y_n - \widehat{y}_n)^2} \tag{7}$$

4. Experimental results and analysis

4.1. The importance of variational ant colony algorithm

This section explains the importance of the variational ant colony algorithm. Three experiments demonstrate the significant advantages of the variational ant colony algorithm, solve the ant colony rank selection problem, and the annealing probability setting problem. This section uses the UK-wide onshore wind power data set. All experiments in section 4.1 was done based on the XGBoost algorithm, and all data in the tables are the average of 100 experiments.

4.1.1. The significant advantages of the variational ant colony algorithm

The experiments were divided into three groups, each group was conducted for six rounds, and 200 parameter combinations were performed in each round. The method used in group I was the Monte Carlo method. The method used in group II was a rank-based ant colony algorithm. The method used in group III was the variational ant colony algorithm. The evaluation metrics are MAE (mean and maximum values), R^2Score (mean and maximum values), super excellent rate (percentage of the number of experiments with R^2Score greater than 0.950 out of the total number of experiments), and excellent rate (percentage of the number of experiments with R^2Score between 0.948 and 0.9499 out of the total number of experiments). The meanings and value ranges of the parameters to be determined

Table 2
Pending parameter meaning and value range table.

Parameter name	Parameter meaning	Parameter initial value range
KDJ-time	A specified period in the FTIs KDJ, used to calculate RSV	3–20
KDJ-w	FTIs KDJ used to calculate the weighting values of K and D	2–6
MACD-f	The calculation period of the fast moving average in the FTIs MACD	2–12
MACD-s	The calculation period of the slow moving average in the FTIs MACD	12–32
MACD-dea	The weighting value used to calculate dea in the FTIs MACD	7–14
RSI-time	A specified period in the FTIs RSI, used to calculate RSI1x	3–9

are shown in Table 2. The parameter optimization performance of the variational ant colony algorithm is shown in Table 3.

As shown in Table 3, the bolded data are the best results, and the underlined data are the second-best results. The evaluation indexes of group II and group III were significantly better than those of group I. The rank-based ant colony algorithm and the parameter optimization method based on the variational ant colony algorithm significantly improved the model prediction accuracy. Although group II achieved a better prediction effect in some evaluation methods, this study requires the best computational parameters to calculate the FTIs. Observe the third and fifth columns (extreme value records) in Table 3. The variational ant colony algorithm can always search for the best FTIs calculation parameters.

Figs. 5, 6 and 7 represent the setting of the calculation parameter MACD-s of the financial technical indicator MACD under different parameter optimization algorithms, respectively, and the red dots in the figures all represent the parameter taking values.

Combining Table 3 with Fig. 5, the search method using only the Monte Carlo method cannot effectively find the optimal computational parameters required for this study. Combining Table 3 with Fig. 6, the combination of Monte Carlo method and rank-based ant colony algorithm results in a significant improvement in the optimal parameter search capability. However, Fig. 6 shows that this method has a local optimal solution problem, and the parameter search range is limited to a specific range.

In combination with Table 3 and Fig. 7, a simulated annealing algorithm is added to the Monte Carlo method combined with the rank-based ant colony algorithm. The evaluation metrics show that the best parameters appear in this group of experiments. It can be seen from Fig. 7 that although a relatively clear parameter search range appeared in the fifth round of experiments, the sixth round of experiments adjusted to expand the search range and solve the local optimal solution problem by benefiting from the addition of the simulated annealing algorithm. Meanwhile, the experimental results show that the parameter combination with the best prediction effect appears in the sixth round of experiments.

4.1.2. Selection of ant colony rank

In variational ant colony algorithms, it is necessary to set a specific "rank" for the rank-based ant colony algorithm. This study conducts five groups of comparison experiments to demonstrate the performance of the variational ant colony algorithm with different "rank". Each group of experiments is conducted for six rounds, with 200 parameter combinations. The parameter ranges corresponding to the top 5, 10, 15, 20, and 30 records of R^2Score in each round was used as the parameter search ranges for the next round of experiments. The results of the experiments are shown in Table 4.

In Table 4, the bolded data are the best results, and the underlined data are the second-best results. Table 4 shows no significant differences between the five experimental groups' R^2Score (mean and maximum values) and MAE (mean and maximum values). However, the two assessment indicators, super excellent rate, and excellent rate, showed a large difference, and the changing trend was ascending and then descending. Combining the above analysis and the experimental results, the best prediction accuracy (the super excellent rate and excellent rate) can be obtained using the parameter range corresponding to the first ten records of R^2Score ranking as the search range for the parameters of the new round of experiments. Therefore, group II's "rank" setting effectively updated the parameter search range and significantly improved the model prediction accuracy. This study also applied this setting to all subsequent experiments.

4.1.3. Setting of annealing probability

In the variational ant colony algorithm, regarding the probabilistic annealing in the simulated annealing algorithm, a specific annealing probability needs to be set to ensure that the parameter search range can be effectively narrowed while solving the local optimal solution problem. In this study, five sets of comparative experiments are conducted to demonstrate the performance of the variational ant colony algorithm under different annealing probabilities. Each set of experiments is conducted for six rounds, with 200 parameter combinations. The annealing probabilities were set to 3%, 5%, 10%, 15%, and 25% for the group I to group V experiments, respectively. The experimental results are shown in Table 5.

In Table 5, the bolded data are the best results, and the underlined data are the second-best results. As shown in Table 5, no significant differences were seen in R^2Score (mean and maximum values) and MAE (mean and minimum values) for the first four groups of experiments. The MAE (mean value) and R^2Score (mean value) of the experiments in group V were significantly lower. This shows that the higher annealing probability is not conducive to the model to improve the prediction accuracy and cannot be used with the rank-based ant colony algorithm to narrow down the parameter search effectively. From the two types of evaluation indexes, the super excellent rate and excellent rate, multiple annealing causes the relevant evaluation indexes to be similar to those of the Monte Carlo method parameter optimization method proposed in this study, resulting in the variational ant colony algorithm losing its significance.

Combining the above analysis and experimental results, group III of experiments (annealing probability set to 10%) achieved better prediction accuracy, effectively narrowed the parameter search range, and solved the local optimal solution problem faced by the

Table 3
Optimization experimental performance table of different algorithm parameters.

Experimental Group	MAE (Mean value)	MAE (Minimal value)	R^2Score (Mean value)	R^2Score (Maximal value)	Super excellent rate	Excellent rate
I	0.7600	0.6794	0.9405	0.9526	0.33%	1.53%
II	**0.7042**	0.6714	**0.9463**	0.9534	**9.20%**	**25.20%**
III	0.7089	**0.6693**	0.9457	**0.9544**	6.60%	18.33%

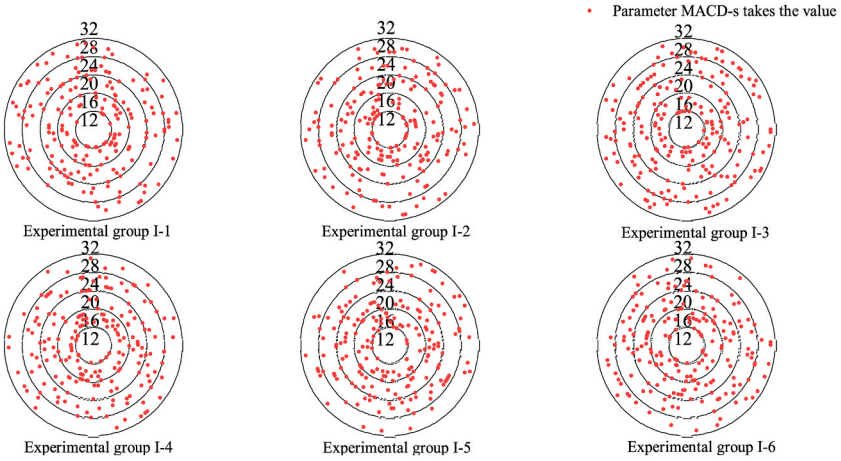


Fig. 5. Schematic diagram of the effect of parameter optimization with MACD-s as an example (Group I).

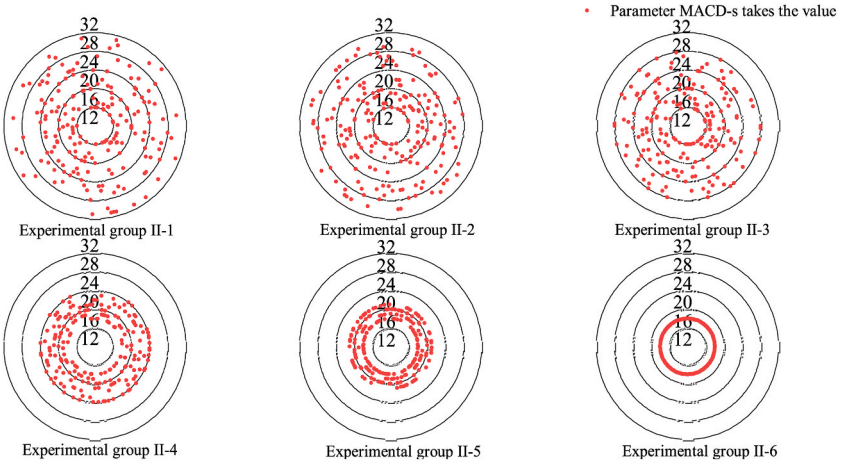


Fig. 6. Schematic diagram of the effect of parameter optimization with MACD-s as an example (Group II).

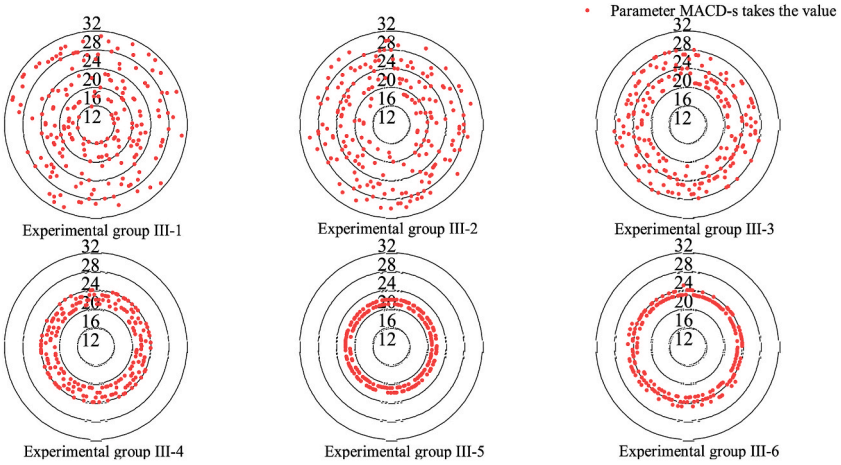


Fig. 7. Schematic diagram of the effect of parameter optimization with MACD-s as an example (Group III).

Table 4
Experimental performance table of different "rank" in variational ant colony algorithm.

Experimental Group	MAE (Mean value)	MAE (Minimal value)	R^2Score (Mean value)	R^2Score (Maximal value)	Super excellent rate	Excellent rate
I	0.7089	**0.6693**	0.9457	**0.9541**	6.60%	18.33%
II	**0.7083**	0.6705	**0.9461**	0.9535	**8.20%**	**22.27%**
III	0.7099	0.6709	0.9458	0.9537	7.30%	21.06%
IV	0.7190	0.6725	0.9449	0.9539	7.26%	15.00%
V	0.7139	0.6713	0.9454	0.9540	5.13%	17.20%

Table 5
Experimental performance table of different annealing probabilities in variational ant colony algorithm.

Experimental Group	MAE (Mean value)	MAE (Minimal value)	R^2Score (Mean value)	R^2Score (Maximal value)	Super excellent rate	Excellent rate	Annealing times
I	**0.7082**	**0.6721**	**0.9460**	0.9530	**7.53%**	**22.13%**	0
II	0.7150	0.6746	0.9453	0.9534	7.27%	21.39%	0
III	0.7160	0.6781	0.9452	**0.9537**	3.60%	15.13%	1
IV	0.7280	0.6748	0.9438	0.9526	2.60%	10.93%	2
V	0.7474	0.6731	0.9415	0.9527	1.47%	5.60%	3

parameter optimization algorithm, and this study also applied this setting to all subsequent experiments.

4.2. Analysis of the validity of FTIs

This section demonstrates the positive impact of FTIs on model prediction accuracy. This section uses a wind power data set from a wind farm in Inner Mongolia, and six groups of experiments are conducted based on the XGBoost algorithm. The data in the table are the average of 100 experiments. The details of the experimental groups are shown in Table 6, and the experimental results are shown in Table 7.

In Table 6, the prediction targets for all experimental groups are the wind power generated in the next 15 min. The input feature of group I is the historical power data. The input features of group II are historical power data with NWP data. The input features of groups III and IV are historical power data and FTIs. The input features of groups V and VI are historical power data, FTIs, and NWP data. In addition, this section additionally adds a comparison experiment for wind power prediction using different FTIs calculation methods. Because groups I and II do not involve FTIs, they do not have FTIs calculation methods. Groups III and V use financial experts' experience to calculate FTIs, while groups IV and VI use variational ant colony algorithms to calculate FTIs.

As shown in Table 7, the bolded data are the best results, and the underlined data are the second-best results. Comparing groups I and II, the MAE is reduced by 9.17%, and the RMSE is reduced by 4.07% when the traditional covariates (e.g., wind speed and wind direction in meteorological data) in the wind power time series data are used as the model input features. The meteorological data slightly enhance the model prediction accuracy. Comparing groups I, III, and IV, the model prediction accuracy is significantly improved with the FTIs as model input features. MAE decreases by 18.33%, and RMSE decreases by 21.11% when FTIs based on financial experts' experience are used as model input features. The MAE is reduced by 22.31%, and the RMSE is reduced by 25.43% when the FTIs based on the variational ant colony algorithm are used as model input features. Comparing groups III and V, groups IV and VI, the model's prediction accuracy did not improve significantly when both meteorological data and FTIs were used as model input features but decreased. This is because adding redundant features reduces the attention of the XGBoost model to important features. Some invalid features are also used to construct the XGBoost model's CART tree.

Further, this study refines the experimental groups III and IV in Table 7. The FTIs calculated by relying on financial experts' experience and those calculated by relying on the variational ant colony algorithm and their prediction effects are shown in Table 8.

As shown in Table 8, the computational parameters of the FTIs derived using the variational ant colony algorithm on the wind power data set of Inner Mongolia, China, are significantly different from those provided by the financial experts' experience. The results show that the variational ant colony algorithm reduces the MAE by about 4.87% and the RMSE by 5.48% in this data set. It can

Table 6
FTIs importance experiment grouping details table.

Experimental Group	Model input features	Calculation method of FTIs parameters	Predictive target
I	historical power	–	next 15 min power
II	historical power, NWP data	–	next 15 min power
III	historical power, FTIs	Financial expert experience	next 15 min power
IV	historical power, FTIs	Variational ant colony algorithm	next 15 min power
V	historical power, FTIs, NWP data	Financial expert experience	next 15 min power
VI	historical power, FTIs, NWP data	Variational ant colony algorithm	next 15 min power

Table 7
FTIs importance experiment results.

Experimental Group	MAE	RMSE	R^2Score
I	2.138	3.146	0.936
II	1.942	3.018	0.942
III	1.746	2.482	0.949
IV	**1.661**	**2.346**	**0.955**
V	1.764	2.533	0.945
VI	1.702	2.446	0.951

Table 8
Parameter optimization effect.

Parameter name	Financial expert experience setting parameters	Variational ant colony algorithm setting parameters
KDJ-time	9	7
KDJ-w	3	2
MACD-f	12	2
MACD-s	26	23
MACD-dea	9	5
RSI-time	6	3
Evaluation metrics	**Financial expert experience setting parameters**	**Variational ant colony algorithm setting parameters**
MAE	1.746	1.661
RMSE	2.482	2.346
R^2Score	0.949	0.955

be seen that the calculation parameters of FTIs matching with the current wind power time series data can effectively improve the model prediction accuracy.

In order to visually represent the prediction differences, this study plots the actual values against the predicted values. As shown in Fig. 8, the black line indicates the actual value, the red line indicates the predicted value after calculating the FTIs using the financial experts' experience, and the blue line indicates the predicted value after calculating the FTIs using the variational ant colony algorithm.

In Fig. 8, the blue line is significantly better fitted than the red line. In the first subplot pointed out by the orange dashed line, we can find that the prediction model with the variational ant colony algorithm always has a good grasp of the trend and magnitude of the power variation in wind power data with short-time fluctuations. In the second subplot pointed out by the orange dashed line, we can see that the prediction model with the variational ant colony algorithm also performs well in smoother wind power data.

Further, on this data set, feature selection experiments were conducted in this study. Three groups of experiments were designed using the FTIs calculated by the variational ant colony algorithm combined with the historical wind power data. The prediction

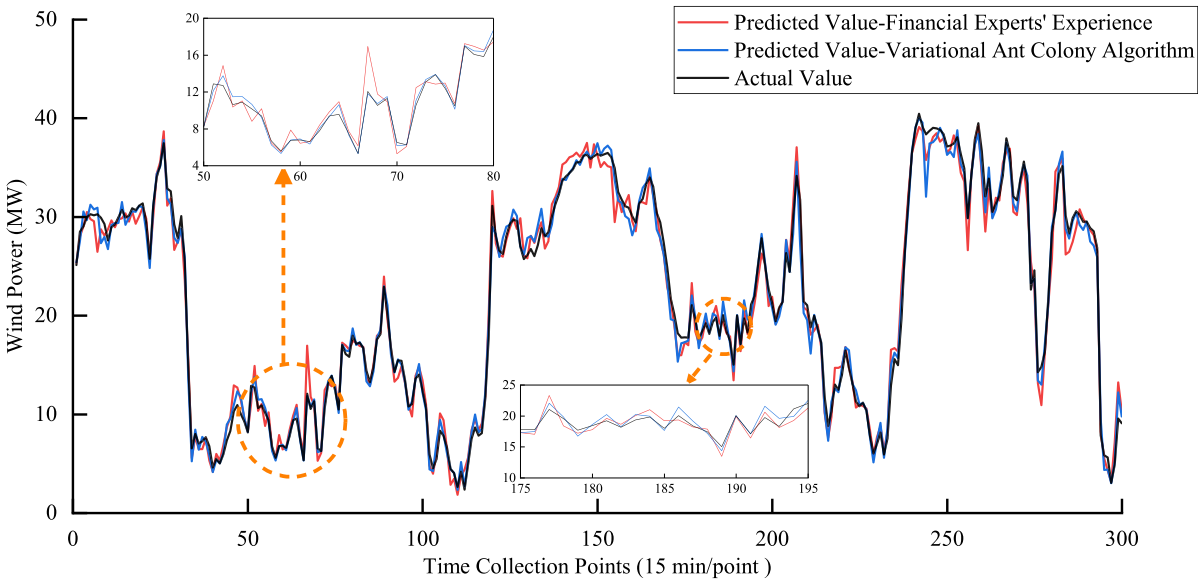


Fig. 8. Predictive effect of FTIs obtained by different methods in the model.

Table 9
Feature selection experiment group details table.

Experimental Group	Model input features	Feature selection method	Predictive target
I	FTIs, historical power	–	next 15 min power
II	FTIs, historical power	PCA	next 15 min power
III	FTIs, historical power	GRA + PCA	next 15 min power

algorithm for three groups of experiments was the XGBoost algorithm. The details of the experimental groups are shown in Table 9, and the experimental results are shown in Table 10, where the data in Table 10 are the average of 100 experiments.

In Table 9, the prediction target for all experimental groups is the wind power generation in the next 15 min. The input features for three sets of experiments are historical power data and FTIs. Group I uses only the XGBoost algorithm's feature selection method. Group II adds the PCA method. Group III adds the GRA method and PCA method.

As shown in Table 10, the bolded data are the best results, and the underlined data are the second-best results. To indicate the complexity of the model, the run time of the model is added in Table 10. This time is the model's total time from reading the data to outputting the prediction results. Three experiments were done on computers with the same environment and configuration. The data in the table indicate that the experimental group that did not perform additional feature selection work achieved the best prediction results. The experimental group that added the PCA method or combined PCA and GRA had poor prediction results. The experimental group that combined PCA and GRA had slightly better prediction results than those that used only the PCA method. Comparing Group I (experimental group without additional feature selection work) with Group III (experimental group combining PCA and GRA), the MAE of Group I is 11.34% lower than the MAE of Group III, and the RMSE of Group I is 4.83% lower than the RMSE of Group III. Looking at the complexity of the model in terms of model running time, the running time of group I is much smaller than that of group II and group III, and adding additional feature selection work algorithms will certainly increase the complexity of the model.

Combining the above analysis with the experimental results in Table 10. The additional feature selection effort did not improve the model prediction accuracy but made it less accurate. The authors believe that there are two reasons for this result. First, during dimensionality reduction, the PCA and GRA methods may have changed the temporal order information and the data's original time series structure. The financial technical indicators MACD, KDJ, and RSI are different from each other in dimension, and normalizing them into the same dimension may easily cause the phenomenon of feature loss. Secondly, the XGBoost algorithm comes with a feature importance assessment function, which can consider the nonlinear relationship between features and the combination effect and interaction effect between features to select the most suitable features for ultra-short-term wind power prediction. Therefore, the features processed by PCA and GRA methods will change or lose some information, which affects the work of the XGBoost algorithm in constructing CART trees so that it cannot consider the combined effect, interaction effect, and nonlinear relationship among features. In this way, the experimental group I, which does not carry additional feature selection work, achieves the best prediction results.

Analysis of several sets of comparison experiments using NWP data as model input features showed that NWP data did not significantly improve model prediction accuracy. The FTIs designed in this study can replace NWP data as the input features of the model. The FTIs feature engineering reduces the dependence of the prediction model on high-precision and high-frequency NWP data. It reduces the economic cost of obtaining such covariates for wind farms. Meanwhile, with the wind power data in Inner Mongolia, we again confirm that the calculated parameters of FTIs derived by the variational ant colony algorithm can effectively improve the model forecasting capability and capture the trend and magnitude of wind power time series data changes. In addition, the experimental results and analysis show that since XGBoost comes with a feature importance assessment method, the prediction model proposed in this study does not need to add additional feature selection methods, such as the PCA method and the GRA method, to obtain the best prediction results. In summary, obtaining FTIs from wind power time series data is an effective and novel feature engineering. Using the variational ant colony algorithm to find the computational parameters of fintech indicators is an effective means to reduce the reliance on the experience of financial experts.

4.3. Comparison experiments between different prediction models

In this section, the German power company Tennet wind power data set is used to verify the accuracy and performance of the model proposed in this study. For information about the data set, see Section 3.1.3. This section uses the LightGBM, Decision Tree, IHBA-SVM, LSTM, TCN, and GRU models to conduct comparative experiments. Use the evaluation indicators mentioned in Section 3.2 and the comparison chart between the actual value and the predicted value to evaluate the prediction effect of the model.

The experiments in this section use the variational ant colony algorithm to search for the best computational parameters of the FTIs, combined with the historical wind power data to calculate the financial technical indicators MACD, KDJ, and RSI. These FTIs are used as the input features of the prediction model along with the historical wind power, and the prediction target is the wind power generation value after 15 min. The details and prediction results of the experimental group are shown in Table 11, and the data in the table are the average of 100 experiments. The bolded data in the table are the best results, and the underlined data are the second-best results. The comparison chart of predicted and actual values is shown in Figs. 9–14. The black line in the figure indicates the actual value, the red line indicates the predicted value of the XGBoost model, and the other colored lines represent the models shown in the legend of each figure.

Comparing Group I (XGBoost model) with Group II (LightGBM model), the MAE of the XGBoost model is 7.34% lower than that of the LightGBM model, and the RMSE of the XGBoost model is 4.04% lower than that of the LightGBM model. In addition, the XGBoost

Table 10
Feature selection experiment results.

Experimental Group	MAE	RMSE	R^2Score	Running Time
I	**1.493**	**2.303**	**0.956**	**0.268s**
II	1.689	2.462	0.950	0.325s
III	1.684	2.420	0.951	0.411s

Table 11
Experimental grouping details and predicted results.

Experimental Group	Prediction model	Model input features	Predictive target	MAE	RMSE	R^2Score	Running time
I	XGBoost (GPU)	historical power, FTIs	next 15 min power	0.859	1.329	0.997	**0.244s**
II	LightGBM	historical power, FTIs	next 15 min power	0.927	1.385	0.994	0.681s
III	Decision tree	historical power, FTIs	next 15 min power	2.327	3.047	0.975	0.905s
IV	IHBA-SVM	historical power, FTIs	next 15 min power	2.397	3.002	0.962	4617s
V	LSTM	historical power, FTIs	next 15 min power	**0.832**	**1.239**	**0.998**	2248s
VI	TCN	historical power, FTIs	next 15 min power	0.941	1.376	0.992	4164s
VII	GRU	historical power, FTIs	next 15 min power	8.019	10.606	0.907	2394s

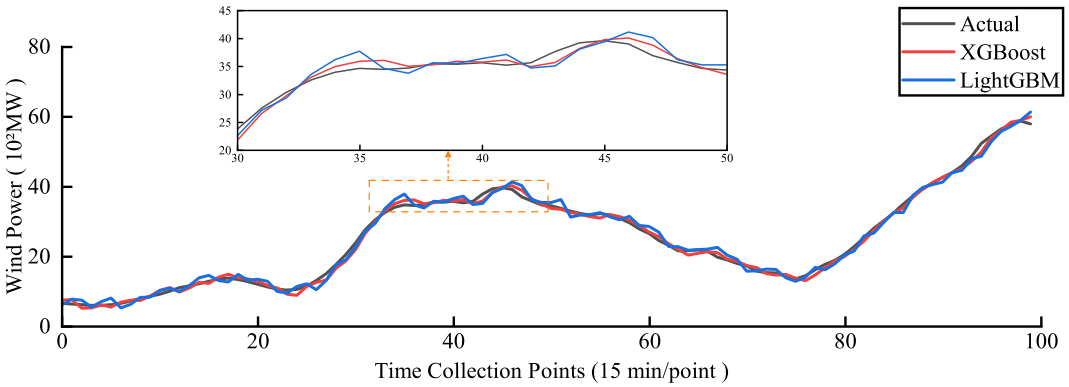


Fig. 9. Comparison of prediction results between the XGBoost and LightGBM models.

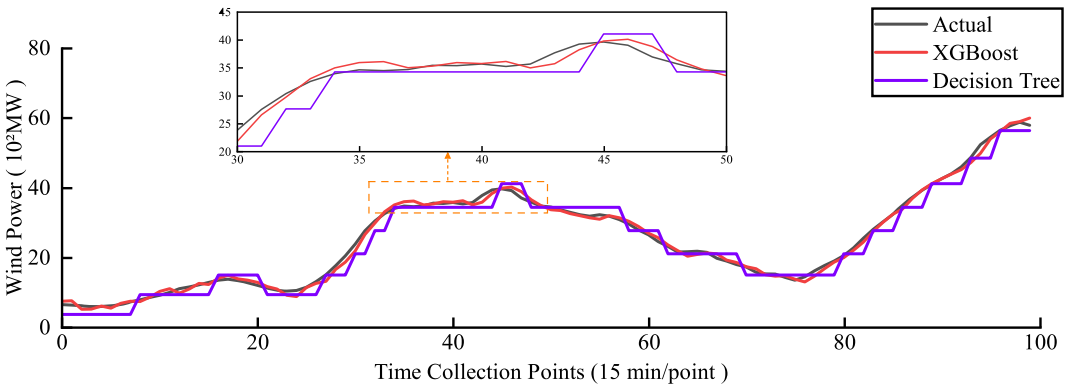


Fig. 10. Comparison of prediction results between the XGBoost and Decision Tree models.

model saves 64.17% of the prediction time.

Observing Fig. 9, the predicted values of the LightGBM model and the XGBoost model are almost identical. In the subplots of Fig. 9, the prediction error of the LightGBM model is slightly higher than that of the XGBoost model at some time collection points. Considering both the prediction accuracy and the prediction task time requirement, the prediction effect of the proposed model in this study is better than that of the LightGBM model.

Comparing Group I (XGBoost model) with Group III (Decision Tree model), the MAE of the XGBoost model is 63.08% lower than that of the Decision Tree model, and the RMSE of the XGBoost model is 56.38% lower than that of the Decision Tree model. In addition,

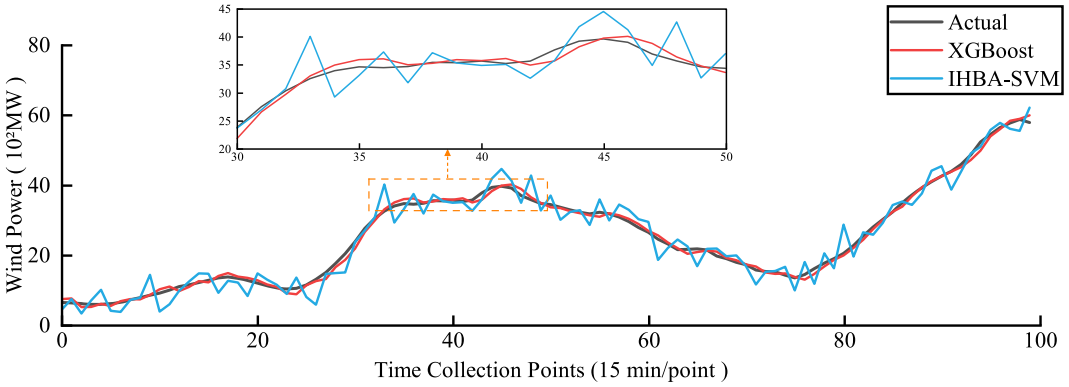


Fig. 11. Comparison of prediction results between the XGBoost and IHBA-SVM models.

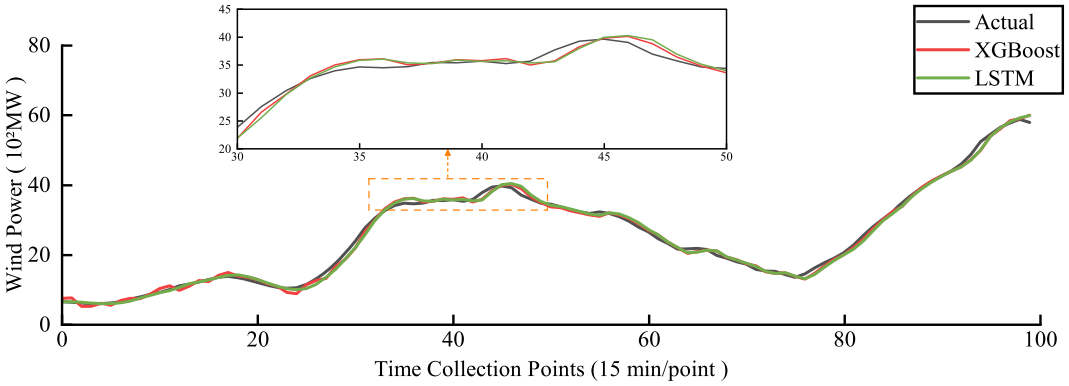


Fig. 12. Comparison of prediction results between the XGBoost and LSTM models.

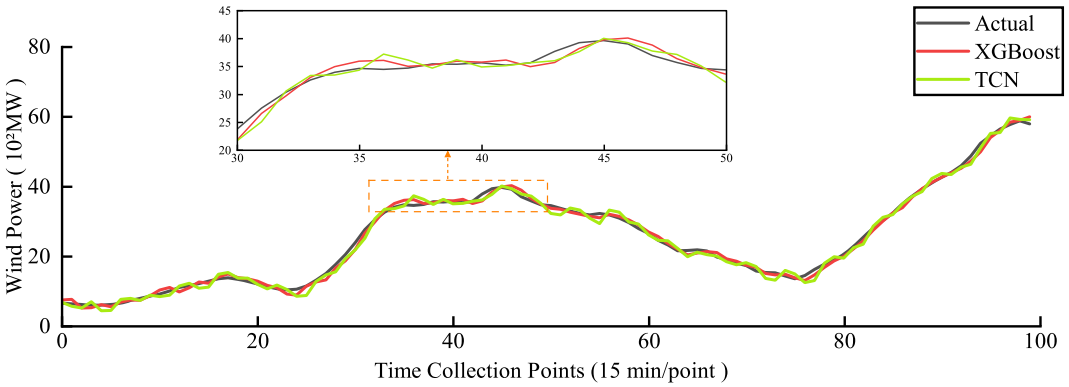


Fig. 13. Comparison of prediction results between the XGBoost and TCN models.

the XGBoost model saves 73.04% of the prediction time.

Observing Fig. 10, the predicted value image of the Decision Tree model is stepped, which can only roughly predict the trend of wind power variation and cannot accurately predict the power value. In the subplot of Fig. 10, the Decision Tree model has a larger prediction error compared to the XGBoost model. Considering both the prediction accuracy and the prediction task time requirement, the prediction effect of the proposed model in this study is better than that of the Decision Tree model.

Comparing Group I (XGBoost model) with Group IV (IHBA-SVM model), the MAE of the XGBoost model is 64.16% lower than that of the IHBA-SVM model, and the RMSE of the XGBoost model is 55.73% lower than that of the IHBA-SVM model. In addition, the IHBA-SVM model requires 75 min for training and prediction, and the ultra-short-term wind power prediction task needs to predict the wind power in the next 15 min, so it does not meet the time requirement of the task. Under the same experimental environment, the training and prediction time of the XGBoost model is much lower than that of the IHBA-SVM model, which can meet the time requirement of the

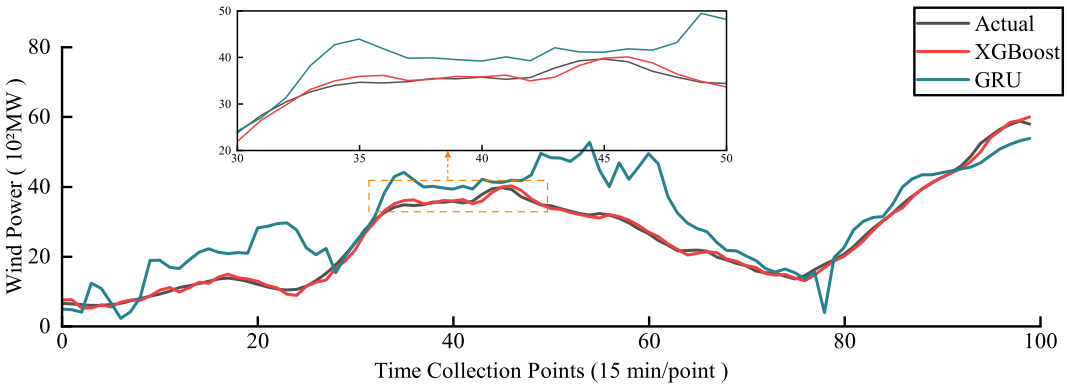


Fig. 14. Comparison of prediction results between the XGBoost and GRU models.

prediction task.

Observing Fig. 11, the prediction images of the IHBA-SVM model are jagged, and the prediction error is close to 30% at some time collection points. In the subplots of Fig. 11, the predicted values of the XGBoost model are significantly closer to the actual values. The predicted values of the IHBA-SVM model fluctuate too much and fail to capture the changing trend of wind power generation. Considering both the prediction accuracy and the prediction task time requirement, the prediction effect of the proposed model in this study is better than that of the IHBA-SVM model.

Comparing Group I (XGBoost model) with Group V (LSTM model), the MAE of the LSTM model is 3.14% lower than that of the XGBoost model, and the RMSE of the LSTM model is 6.77% lower than that of the XGBoost model. In addition, the LSTM model requires 35 min for training and prediction, and the ultra-short-term wind power prediction task needs to predict the wind power in the next 15 min, so it does not meet the time requirement of the task. Under the same experimental environment, the training and prediction time of the XGBoost model is much lower than that of the LSTM model, which can meet the time requirement of the prediction task.

Observing Fig. 12, the predicted values of the LSTM model are similar to those of the XGBoost model, and the predicted values of both models are basically consistent with the actual values, which can accurately grasp the changing trend of wind power generation. Considering the prediction accuracy, the prediction effect of the LSTM model is slightly better than that of the model proposed in this study. Further considered in conjunction with the prediction time, the LSTM model requires too long modeling and prediction time and lacks usability. The model proposed in this study still has the best prediction results under the premise of meeting the time requirement of the prediction task.

Comparing Group I (XGBoost model) with Group VI (TCN model), the MAE of the XGBoost model is 8.71% lower than that of the TCN model, and the RMSE of the XGBoost model is 3.42% lower than that of the TCN model. In addition, the TCN model requires 70 min for training and prediction, and the ultra-short-term wind power prediction task needs to predict the wind power in the next 15 min, so it does not meet the time requirement of the task. Under the same experimental environment, the training and prediction time of the XGBoost model is much lower than that of the TCN model, which can meet the time requirement of the prediction task.

Observing Fig. 13, the predicted values of the TCN model and the XGBoost model are almost the same, the prediction curves of the two models overlap, and the prediction effects of the two models are similar from the comparison graph the actual and predicted values. However, considering both the MAE and RMSE analysis and the prediction task time requirement, the prediction effect of the proposed model in this study is better than that of the TCN model.

Comparing Group I (XGBoost model) with Group VII (GRU model), the MAE of the XGBoost model is 89.29% lower than that of the GRU model, and the RMSE of the XGBoost model is 87.47% lower than that of the GRU model. In addition, the GRU model requires 40 min for training and prediction, and the ultra-short-term wind power prediction task needs to predict the wind power in the next 15 min, so it does not meet the time requirement of the task. Under the same experimental environment, the training and prediction time of the XGBoost model is much lower than that of the GRU model, which can meet the time requirement of the prediction task.

Observing Fig. 14, the prediction error of the GRU model is the highest among all the compared models. The predicted value of the GRU model is far from the actual value and cannot grasp the changing trend of wind power generation. Considering both the prediction accuracy and the prediction task time requirement, the prediction effect of the proposed model in this study is much better than that of the GRU model.

Groups II-IV used comparative models based on machine learning algorithms. The analysis of MAE and RMSE and the comparison graph of the actual and predicted values (Figs. 9–11) show that the model proposed in this study has the highest prediction accuracy. Group V-VII used a comparison model based on deep learning (neural network) algorithm. From the analysis of MAE and RMSE coming and the comparison graphs of actual and predicted values (Figs. 12–14), the prediction effect of the LSTM model shows a competitive advantage compared with the model proposed in this study, the prediction effect of the TCN model is similar to the model proposed in this study. The prediction effect of the GRU model is not as good as the model proposed in this study. In general, the proposed model outperforms the common machine learning models in terms of prediction accuracy and can be comparable to or even exceed the common deep learning models.

In addition, ultra-short-term wind power needs to predict the wind power values for the next 15 min. For the dispatching center to adjust the distribution plan and the trading center to make power futures trading strategies on time, the model construction and prediction time must be much less than 15 min. At the same time, considering that short-term data greatly impacts FTIs, the model needs to be re-modeled for each prediction, so the model construction and prediction time will no longer have application value if it exceeds 15 min. As can be seen from Table 11, the construction time of deep learning (neural network) models is much longer than 15 min, and some models even take more than 60 min to build. The LSTM model with the most competitive prediction accuracy takes 35 min to complete the construction and prediction. In this way, deep learning cannot complete the prediction task in the required time, making it unsuitable for application to power prediction for ultra-short-term wind power generation. Further, the XGBoost model used in this study takes only 244 ms to complete modeling and prediction, which meets the prediction time requirement for ultra-short-term wind power generation and has the possibility of application in wind farms or data centers.

In summary, the model proposed in this study has a very fast prediction speed, which can match or even exceed the prediction accuracy of deep learning models, and has greater application prospects as an effective algorithm for ultra-short-term wind power prediction. In addition, this study carefully designed the FTIs based on financial technology and developed the optimization strategy of calculation parameters corresponding to it, which can provide an effective basis for wind power prediction for wind farms lacking NWP data, and can also provide support for regional prediction (Generally speaking, NWP data for a region cannot be used for wind power prediction, and NWP data is only valid for individual turbine output prediction). Overall, this model provides a new solution for ultra-short-term wind power prediction.

5. Discussion

5.1. The role of FTIs in this study

Financial time series data and wind power time series data have similarities and correlations. The FTI based on wind power time series data uses the start, end, maximum, and minimum values of wind power data over a short period, combined with mathematical theorems (e.g., momentum theory, moving average theory, etc.) perform the calculation. The authors consider this process similar to constructing a long and short-term memory network. The FTI algorithm is similar to a long and short-term memory network with a given calculation method and period. Once the period and computation parameters are specifically set, the computation speed of the features will increase significantly. The modeling and prediction time is significantly reduced by the carefully designed feature engineering with the XGBoost algorithm to build the prediction model.

In addition, using FTIs to predict stock movements is a common approach in financial markets. However, the migration of fintech indicators to wind power time series data is relatively new in the current study, and using fintech indicators to predict future wind power is innovative. The experiments in this study show that the model with FTIs has higher prediction accuracy than the model with NWP data. FTIs can be used as input features of the prediction model instead of NWP data. At the same time, the calculation of FTIs relies only on historical wind power data and does not require NWP data. This technique can eliminate the need for NWP data by wind farms and thus reduce the operating costs of wind farms.

5.2. The role of variational ant colony algorithm in this study

Although financial time series data and wind power time series data have similarities, they have some differences. This study points out that FTIs based on wind power time series data need to set many parameters (e.g., KDJ-time, KDJ-w, MACD-f, etc.) in the calculation. Since the study's goal is ultra-short-term wind power prediction, setting parameters cannot rely on financial experts' experience and requires using parameters more consistent with the wind power variation pattern. In this context, the variational ant colony algorithm is designed in this study to solve this problem.

The variational ant colony algorithm combines the Monte Carlo method, rank-based ant colony algorithm, and simulated annealing algorithm. The Monte Carlo method is a valuation method that can deal with nonlinear and large fluctuation problems, and the property is very consistent with the wind power variation law. The rank-based ant colony algorithm can effectively narrow the parameter search range and improve the parameter search efficiency. The simulated annealing algorithm can prevent the phenomenon of locally optimal solutions of parameters and, at the same time, alleviate the cumulative effect caused by the rank ant colony algorithm. Multiple sets of experiments in this study demonstrate that the variational ant colony algorithm can find the optimal parameters for the computation of FTIs. The FTIs calculated based on the variational ant colony algorithm can give the model higher prediction accuracy than the FITs calculated based on expert experience.

6. Conclusion

Wind energy is one of the more important renewable energy sources, and wind power generation is an important initiative to cope with the climate crisis and energy security. With the global development of wind power construction, accurate prediction of ultra-short-term wind power generation is particularly important. This is because it is crucial to planning power dispatch, buying and selling power futures, and planning the power generation capacity of power farms. Fast and accurate forecasting of ultra-short-term wind power will provide a strong guarantee for the above work. Therefore, this paper proposes an ultra-short-term wind power prediction model based on the XGBoost algorithm combined with financial technology. In addition, a parameter search method based on the variational ant colony algorithm for calculating FTIs is also proposed. Fast and accurate wind power prediction can be achieved

using the variational ant colony algorithm to search for the best calculation parameters of FTIs and using the FTIs and historical wind power data as the model's input.

In order to effectively analyze the validity and reliability of the proposed model, this study carried out wind power prediction using three real wind power data sets. In addition, the proposed model is compared with LightGBM, Decision Tree, IHBA-SVM, LSTM, TCN, and GRU models. The experimental results show that:

- Financial technology feature engineering is an effective feature engineering in the field of wind power prediction, which can construct multiple FTIs to represent the potential relationships among wind power data. This engineering reduces the reliance of wind power prediction models on NWP data and effectively improves model prediction accuracy.
- The variational ant colony algorithm can adaptively find the parameters for calculating FTIs based on wind power time series data, reduce the reliance on financial experts' experience, and make the FTIs more consistent with the variation pattern of wind power time series data.
- The proposed model in this study is based on the XGBoost algorithm, and the proposed model has the fastest prediction speed among all the comparison models. Regarding accuracy, it outperforms all the machine learning models in the comparison models and can match or even exceed the deep learning models. Since the XGBoost algorithm does not require high computing power for hardware support, the model can be deployed to wind farms for real-time prediction and put into wind farm scheduled operation.

In conclusion, this study provides a new solution for ultra-short-term wind power prediction. The calculation of FTIs does not depend on NWP data, so an important contribution of this study is to provide a power prediction solution for wind farms that lack NWP data and ideas for regional wind power prediction. In addition, highly accurate NWP data require a high price to obtain. Since the FTIs proposed in this study are calculated without relying on NWP data, wind farms will save some operating costs when using this model to predict wind power.

At the same time, there are some shortcomings and limitations due to this study. The integrity of wind power data directly affects the calculation of FTIs. However, due to various reasons, there are outliers and missing values in wind power data, and proper handling of outliers and missing values will further enhance the model's prediction accuracy. In addition, MACD, KDJ, and RSI are all short-term FTIs, which do not apply to long-term wind power prediction. Further exploration of the role of FTIs for long-term wind power prediction would benefit the development of wind power generation. Therefore, the outlook of this study for future work is as follows:

- The use of outlier detection and missing value interpolation techniques to improve the reliability and integrity of the data may be a meaningful study, which will potentially further improve the model's prediction accuracy. However, attention should be paid to anomaly detection and missing interpolation efficiency, ensuring that the model still meets the prediction time requirements for ultra-short-term wind power after these works are added. A prediction model that does not meet the time requirements will be meaningless.
- Medium and long-term wind power prediction using medium and long-term FTIs. Medium and long-term wind power predictions are useful for assessing the location of wind farms, maintenance and repair of wind turbines, and planning and evaluating wind power projects.
- The authors argue that there is room for further research in FTIs. Considering "wind speed" as "trading volume" in financial markets, more FTIs can be obtained, and these FTIs may further improve model prediction accuracy, but further experiments and proofs are needed.

Author contribution statement

Shijie Guan: Conceived and designed the experiments; Performed the experiments; Analyzed and interpreted the data; Contributed reagents, materials, analysis tools or data; Wrote the paper.

Yongsheng Wang; Sijia Kan: Analyzed and interpreted the data; Contributed reagents, materials, analysis tools or data; Wrote the paper.

Limin Liu; Jing Gao; Zhiwei Xu: Contributed reagents, materials, analysis tools or data; Wrote the paper.

Data availability statement

Data will be made available on request.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

All authors thank the editors and reviewers for their willingness to take the time and effort to review this manuscript with patience. We thank the editors and reviewers for their valuable comments. Comments made by the editor and reviewers significantly enhanced

the quality of this manuscript.

This study was supported by the National Natural Science Foundation of China [Project No. 61962045], the Natural Science Foundation of Inner Mongolia Autonomous Region [Project No. 2021LHMS06001 and Project No. 2019MS03014], Key Technologies R&D Program of Inner Mongolia Autonomous Region [Project No. 2020GG0094].

References

[1] A. Gopi, P. Sharma, K. Sudhakar, W.K. Ngui, I. Kirpichnikova, E. Cuce, Weather impact on solar farm performance: a comparative analysis of machine learning techniques, Sustainability 15 (1) (2023) 439.
[2] Global Wind Energy Council, Global Wind Report 2023, 2023. https://gwec.net/globalwindreport2023/.
[3] C. Croonenbroeck, C.M. Dahl, Accurate medium-term wind power forecasting in a censored classification framework, Energy 73 (2014) 221–232.
[4] W. Zhang, Z. Lin, X. Liu, Short-term offshore wind power forecasting-A hybrid model based on discrete wavelet transform (DWT), seasonal autoregressive integrated moving average (SARIMA), and deep-learning-based long short-term memory (LSTM), Renew. Energy 185 (2022) 611–628.
[5] F. Zhang, P.C. Li, L. Gao, Y.Q. Liu, X.Y. Ren, Application of autoregressive dynamic adaptive (ARDA) model in real-time wind power forecasting, Renew. Energy 169 (2021) 129–143.
[6] H.B. Ouyang, K. Huang, H. Yan, Prediction of financial time series based on LSTM neural network, Chin. J. Manag. Sci. 28 (4) (2020) 27–35.
[7] C. Gupta, I. Johri, K. Srinivasan, Y.C. Hu, S.M. Qaisar, K.Y. Huang, A systematic review on machine learning and deep learning models for electronic information security in mobile networks, Sensors 22 (5) (2022) 2017.
[8] K. Shi, Y. Qiao, W. Zhao, Q. Wang, M. Liu, Z. Lu, An improved random forest model of short-term wind-power forecasting to enhance accuracy, efficiency, and robustness, Wind Energy 21 (12) (2018) 1383–1394.
[9] Y. Ju, G. Sun, Q. Chen, M. Zhang, H. Zhu, M.U. Rehman, A model combining convolutional neural network and LightGBM algorithm for ultra-short-term wind power forecasting, IEEE Access 7 (2019) 28309–28318.
[10] C. Sasser, M. Yu, R. Delgado, Improvement of wind power prediction from meteorological characterization with machine learning models, Renew. Energy 183 (2022) 491–501.
[11] J. Dong, W. Zeng, L. Wu, J. Huang, T. Gaiser, A.K. Srivastava, Enhancing short-term forecasting of daily precipitation using numerical weather prediction bias correcting with XGBoost in different regions of China, Eng. Appl. Artif. Intell. 117 (2023), 105579.
[12] S. Farah, N. Humaira, Z. Aneela, E. Steffen, Short-term multi-hour ahead country-wide wind power prediction for Germany using gated recurrent unit deep learning, Renew. Sustain. Energy Rev. 167 (2022), 112700.
[13] Y. Zhang, Y. Li, G. Zhang, Short-term wind power forecasting approach based on Seq2Seq model using NWP data, Energy 213 (2020), 118371.
[14] Y. Chengqing, Y. Guangxi, Y. Chengming, Z. Yu, M. Xiwei, A multi-factor driven spatiotemporal wind power prediction model based on ensemble deep graph attention reinforcement learning networks, Energy 263 (2023), 126034.
[15] R. Yuan, B. Wang, Z. Mao, J. Watada, Multi-objective wind power scenario forecasting based on PG-GAN, Energy 226 (2021), 120379.
[16] L.L. Li, Y.B. Chang, M.L. Tseng, J.Q. Liu, M.K. Lim, Wind power prediction using a novel model on wavelet decomposition-support vector machines-improved atomic search algorithm, J. Clean. Prod. 270 (2020), 121817.
[17] M. Liu, K. Luo, J. Zhang, S. Chen, A stock selection algorithm hybridizing grey wolf optimizer and support vector regression, Expert Syst. Appl. 179 (2021), 115078.
[18] C. Li, S. Lin, F. Xu, D. Liu, J. Liu, Short-term wind power prediction based on data mining technology and improved support vector machine method: a case study in Northwest China, J. Clean. Prod. 205 (2018) 909–922.
[19] S. Elsayed, D. Thyssens, A. Rashed, H.S. Jomaa, L. Schmidt-Thieme, Do We Really Need Deep Learning Models for Time Series Forecasting?, 2021 arXiv preprint arXiv:2101.02118.
[20] H. Yuwen, Stock forecast based on optimized LSTM model, Comp. Sci. (S1) (2021) 151–157.
[21] X.M. Li, S.Q. Zhang, Overview of some optimization algorithm based on bionic theory [j], Appl. Res. Comput. 26 (6) (2009) 2032–2034.
[22] W.A.N.G. Bo, L.I.U. Chun, J. Zhang, Uncertainty evaluation of wind power prediction based on Monte-Carlo method, High Volt. Eng. 41 (10) (2015) 3385–3391.
[23] T. Zhang, C. Yu, Y. Zhang, W. Tian, Ant colony system based on the asrank and mmas for the vrpspd, 2007, September, in: International Conference on Wireless Communications, Networking and Mobile Computing, 2007, pp. 3728–3731 (IEEE).
[24] Z.C. Yan, Y.S. Luo, A particle swarm optimization algorithm based on simulated annealing, 989, in: Advanced Materials Research, Trans Tech Publications Ltd, 2014, pp. 2301–2305.
[25] L.P. Joseph, R.C. Deo, R. Prasad, S. Salcedo-Sanz, N. Raj, J. Soar, Near Real-Time Wind Speed Forecast Model with Bidirectional LSTM Networks, Renewable Energy, 2023.
[26] M. Atiquzzaman, N. Yen, Z. Xu (Eds.), Big Data Analytics for Cyber-Physical System in Smart City: BDCPS 2019, December 2019, Shenyang, China, vol. 1117, Springer Nature, 2020, pp. 28–29.
[27] Securities & Futures of China, 2012;07:14. Hongtao, Z., Application of KDJ Index in Security Investment Analysis, Secur. Futur. China 7 (2012) 14.
[28] S. Diange, On the flying safety risk prediction and forecast of the civilian airports based on the MACD index, J. Saf. Environ. 5 (2021) 1911–1918.
[29] R. Rajamoorthy, G. Arunachalam, P. Kasinathan, R. Devendiran, P. Ahmadi, S. Pandiyan, P. Sharma, A novel intelligent transport system charging scheduling for electric vehicles using Grey Wolf Optimizer and Sail Fish Optimization algorithms, Energy Sources, Part A Recovery, Util. Environ. Eff. 44 (2) (2022) 3555–3575.
[30] H. Zheng, Y. Wu, A xgboost model with weather similarity analysis and feature engineering for short-term wind power forecasting, Appl. Sci. 9 (15) (2019) 3019.
[31] P. Sharma, B.J. Bora, A review of modern machine learning techniques in the prediction of remaining useful life of lithium-ion batteries, Batteries 9 (1) (2022) 13.
[32] J.R. Ban, Q. Gou, Y.S. Li, Study on rainfall prediction of yibin city based on GRU and XGBoost, in: 2022 4th International Conference on Advances in Computer Technology, Information Science and Communications (CTISC), IEEE, 2022, April, pp. 1–5.
[33] W. Guilan, Z. Hongshan, M. Zengqiang, Application of XGBoost algorithm in prediction of wind motor main bearing fault, Electr. Power Autom. Equip. 39 (1) (2019) 73–77.
[34] Z. Said, P. Sharma, B.J. Bora, A.K. Pandey, Sonication impact on thermal conductivity of f-MWCNT nanofluids using XGBoost and Gaussian process regression, J. Taiwan Inst. Chem. Eng. 145 (2023), 104818.
[35] L. Bo, Q. Chuan, J. Ping, Short-term bus load forecasting based on XGBoost and Stacking model fusion, Electr. Power Autom. Equip. 40 (3) (2020) 147–153.