Earth Science Informatics (2025) 18:333
https://doi.org/10.1007/s12145-025-01851-8

RESEARCH

Short-term wind speed prediction model based on long short-term memory network with feature extraction

Zhongda Tian[1] · Xiyan Yu[1] · Guokui Feng[1]

Received: 11 December 2024 / Accepted: 13 March 2025

Abstract
For improving the forecast precision of short-term wind speed, this paper proposes a short-term wind speed prediction model based on long short-term memory network with feature extraction. Firstly, for reducing the uncertainty and complexity of short-term wind speed, data cluster is carried out by K-means clustering algorithm. Then, for each clustered sample set, a convolutional neural network is used to further characterize the short-term wind speed, making the features of the wind speed data more regular. Finally, the short-term wind speed data after feature extraction is predicted using long short-term memory network. Meanwhile, an improved Whale optimization algorithm is proposed to optimize the learning rate, regularization coefficient, and number of hidden layer nodes of the long short-term memory network. The accuracy of the prediction model is tested by using the actual short-term wind speed data with sampling time of 5 min and 30 min. The comparative analysis of relevant performance indicators shows that compared with other models, the established prediction model has good prediction performance and can well reflect the characteristics of short-term wind speed changes. The designed model is effective in improving the accuracy of short-term wind speed prediction.

Keywords Short-term wind speed prediction · K-means clustering · Convolutional neural network · Long short-term memory network · Improved whale optimization algorithm

Introduction

Background

Energy is an important pillar of economic and social development and progress. Today, the global energy supply crisis is becoming increasingly severe. The international energy supply and demand situation is complex and severe, with significant increases in energy prices. Therefore, the development and application of new energy is urgently needed. New energy mainly includes solar energy, biomass energy, hydro energy, wind energy, geothermal energy, etc. Wind energy stands out among many new energy sources due to its mature development technology, broad commercial prospects, and strong stability. Wind power generation, as a clean and renewable form of energy, is gradually receiving high attention and importance from countries around the world. The primary factor in wind power generation is whether the wind speed meets the requirements for wind power generation. Therefore, wind speed prediction is crucial for wind power generation (Zhang et al. 2024).

According to the time scale, wind speed prediction can be divided into ultra-short-term wind speed prediction, short-term wind speed prediction, medium-term wind speed prediction, and long-term wind speed prediction. The specific time scale and purpose of wind speed prediction are shown in Table 1. The time required for power grid scheduling and resource allocation is generally 0 to 3 h. Therefore, short-term wind speed prediction is of great significance for power balance and operation scheduling of the power grid.

Research status

In recent years, research on wind speed prediction has received widespread attention and development. The methods of wind speed prediction are constantly improving, and the accuracy and speed of prediction are also constantly

Communicated by: Hassan Babaie.

✉ Zhongda Tian
tianzhongda@126.com

1 School of Artificial Intelligence, Shenyang University of Technology, Shenyang 110870, China

Published online: 21 March 2025

Table 1 Classification and application of wind speed prediction

Classification	Time scale	Application
Ultra-short- term	0 ~ 30 min	Prevention and control of wind turbines
Short-term	30 ~ 360 min	Develop a power grid dispatch plan to ensure stable power transmission
Medium-term	Over a week	Wind turbine inspection and maintenance
Long-term	Over a year	Feasibility assessment of site selection and construction of wind farms

improving. The technology of wind speed prediction is also becoming more and more mature. After years of development, the methods for wind speed prediction can be divided into the following categories: physical method, statistical method, machine learning method, combination method, etc.

Physical method

In the 1990s, a physical model-based wind speed prediction system was developed, which integrates various factors closely related to wind speed such as geographical location, temperature, and altitude to construct a prediction model. A classic category of physical modeling methods is numerical weather prediction (NWP), which requires a large amount of information for its construction. In addition to weather data, it also includes wind farm site selection terrain, surrounding obstacles, and even digital ground models (Hoolohan et al. 2018). The NWP method is difficult to meet the real-time and efficient requirements of short-term wind speed prediction in power systems due to its large amount of information and complex calculations (Valdivia-Bautista et al. 2023).

Statistical method

Statistical methods require the use of historical wind speed time series data, and when the amount of observed data is small, it can affect prediction accuracy. Compared with physical methods, statistical methods do not require any physical information collection from wind farms to construct prediction models, requiring less computational resources and time. They mainly focus on the sustained characteristics of wind to achieve prediction. By searching for the characteristics between wind speeds, finding patterns and using them as a basis to establish a prediction model. The commonly used models include auto regressive moving average (ARMA) model and auto regressive integrated moving average (ARIMA) model. In the literature (Jaeseok et al. 2016), the author used ARMA as a basis for short-term wind speed prediction. In the literature (Aasim et al. 2019), the author used the ARIMA model to predict short-term wind speed. The wind speed data is characterized by volatility and randomness, and a single model cannot adapt to this change. Only historical wind speed data is used to train the model, and the model training parameters do not meet the prediction effect. The accuracy of statistical methods is not high, that is, statistical methods cannot accurately describe nonlinear data. Statistical methods are typically suitable for short-term wind speed prediction.

Machine learning method

With the development of computers, machine learning technology is becoming increasingly advanced, and its application in short-term wind speed prediction is becoming more and more common. Machine learning methods are generally divided into neural networks, support vector machines, deep learning models, etc.

The pattern of using neural networks to extract features is gradually being understood and applied in the field of wind speed prediction. Tan et al. proposed a prediction model that utilizes the salp swarm algorithm to optimize the input weights and biases of hidden nodes in Extreme Learning Machine. The research results indicate that this method effectively avoids the phenomenon of the model falling into local minima, and has faster convergence speed and higher accuracy (Tan et al. 2020). Cui et al. studied a ultra-short-term wind speed prediction method that combines chaos theory and artificial bee colony algorithm to optimize radial basis function (RBF) neural network. The experimental results show that the proposed method has good accuracy and robustness (Cui et al. 2019). Zucatelli et al. studied the most effective artificial neural network configuration for predicting wind speed, which can effectively determine the optimal position of wind turbines and accurately predict wind speed at different wind measurement heights (Zucatelli et al. 2019a). Zucatelli et al. studied the application of different neural network configurations at different locations and heights, conducted quantitative analysis, and evaluated the statistical results to select the configuration that best predicts real data. The results indicate that Multilayer Perceptron is the optimal configuration for short-term wind speed prediction (Zucatelli et al. 2019b). The neural networks prediction method expresses the correlation between independent and dependent variables through the weights between neurons, and its structure can handle real-time problems well with good fault tolerance. Neural network can be used for regression prediction of massive information samples, but there are also problems such as slow learning speed and local optimal behavior.

Support vector machines establish a functional mapping between independent and dependent variables by analyzing historical data, and predict unknown data based on this mapping. Li et al. proposed a dragonfly algorithm that combines adaptive learning factors and differential evolution strategy to select the optimal parameters for support vector machines (SVM), which is susceptible to parameter settings and significantly improves prediction performance (Li et al. 2020). Li et al. addressed the issue of poor prediction accuracy of a single least squares support vector machine (LSSVM) in wind speed prediction, and combined an improved ant colony algorithm with LSSVM to form a new prediction model. The experiment proves that this method has high generalization ability (Li et al. 2019b. The advantage of SVM and LSSVM is that they have good predictive performance for small data samples, but the disadvantage is that the selection of kernel function parameters and penalty factors is difficult, and intelligent optimization algorithms are usually needed to determine these parameters, which in turn increases the complexity of the algorithm.

In recent years, the deep learning model based on the traditional neural network has also been applied to the prediction of wind speed. These deep learning forecasting models include CNN (Duan et al. 2022), gated recurrent unit (Wang et al. 2024a), LSTM (Wang et al. 2023), Transformer (Yu et al. 2024), etc. However, the deep learning training needs the support of a large number of sample data, and the training process occupies a lot of computing resources. This requires research on how to reduce the number of input samples and decrease the complexity of operations.

Combination method

Through a large number of studies, it is found that it is challenging to use a single model to predict the ultra-short-term wind speed, and for the wind speed information of different wind turbines in different time periods, the prediction results of a single model are very different. Due to the limited forecasting ability of a single model, more and more researchers begin to pay attention to the combination forecasting model. The combination prediction model combines the advantages of each method, improves the accuracy of prediction, and facilitates the selection process. Due to their excellent performance in predicting ultra-short-term wind speed, these models have been widely adopted (Zhang et al. 2023).

Zucatelli et al. proposed a supervised prediction method based on deep learning and wavelet decomposition for wind speed prediction at different heights of wind farms. The results indicate that the method has satisfactory performance in predicting wind power generation and wind power slope in tropical and subtropical regions of South America (Zucatelli et al. 2021). Zucatelli et al. proposed a method for predicting wind speed at different heights and locations based on artificial neural networks and wavelet decomposition techniques. Research has shown that the best predictive models are recurrent neural networks and Meyer wavelets, and the proposed hybrid model can effectively predict wind speed (Zucatelli et al. 2020). Yang et al. developed a wind speed prediction method based on improved Singular Spectrum Analysis data decomposition. The research results have demonstrated the effectiveness of the established prediction method (Yang et al. 2022). However, the combination prediction model often has a complex structure, which increases the training time and has certain limitations in its application in short-term wind speed prediction.

Contributions of the paper

Wind speed has the characteristics of nonlinearity, volatility, and susceptibility to meteorological factors, which makes wind speed prediction challenging. Although many prediction models have been proposed, predicting wind speed has always been a challenge. Many short-term wind speed prediction models currently proposed often focus on the construction or optimization of the model itself, while ignoring the uncertainty of input data and the inherent features. For future wind speeds, the wind speed values at different sampling times in history have different importance and correlation, that is, the impact of different historical wind speeds on future wind speed changes is different. The K-means algorithm clusters wind speed samples, grouping wind speeds with similar characteristics into one cluster, and then selecting an appropriate model is beneficial for improving prediction performance.

The local features within the clustered wind speed samples are also worth analyzing. Convolutional neural network (CNN) is suitable for extracting local features. CNN is used to extract features from each cluster sample set. Long short-term memory network (LSTM) can adapt to input sequences of different lengths and effectively handle the temporal dependencies of data, making them suitable for processing time series data. Therefore, LSTM is used to predict CNN processed data. Furthermore, the hyper-parameters of LSTM have a significant impact on its performance. This paper proposes an improved Whale optimization algorithm (IWOA) to optimize LSTM and further improve the accuracy of the model. For the wind speed data samples to be predicted, after K-means clustering calculation, the cluster to which they belong is obtained, and then the corresponding CNN-LSTM model is selected to obtain the predicted value. The output values of multiple LSTMs are sorted and adjusted to form the final prediction results. The research object of this study is a single wind turbine, which does not require spatial feature extraction. Therefore, the CNN used in the paper is one dimensional convolutional neural network (1D-CNN). The main contributions of this study are as follows.

1. By using the K-means algorithm to cluster the original short-term wind speed sequence, the uncertainty and

volatility of the original data can be reduced, which can better extract the initial features of the wind speed sequence.
2. The CNN-LSTM model utilizes the local features extraction capability of CNN and the long-term temporal relationship extraction capability of LSTM in data.
3. An IWOA with better performance is proposed to optimize the hyper-parameters of LSTM.

Structure of the paper

The remainder of the paper is structured as follows: Section "Data sets" introduces the short-term wind speed data set and some data indicators used in the research. Section" Proposed prediction model" mainly introduces the principle and implementation process of the proposed prediction model. Section "Materials and methods" underlying theories of K-means algorithm, CNN, LSTM and IWOA. Section "Experiment and results" presents the experimental results and error analysis. Section "Conclusions" provides a summary and future work of the entire paper.

Data sets

The two data sets in this study are collected from Manjing Wind Farm in Shangyi County, Zhangjiakou City, Hebei Province, People's Republic of China. Due to limitations in sensor facilities, only wind speed data is collected, while other data such as temperature, atmospheric pressure, and wind direction, etc. are not collected. After data collection, it is stored in the database in a way that corresponds to the collection time and wind speed values. When generating sample data, extract data from the database at sampling intervals to generate wind speed data sets at different time scales. This study generated a short-term wind speed data set based on 5-min and named it Data set 1. Another short-term wind speed dataset is generated at 30-min intervals and named Data set 2. The data volume of both datasets is 4300.

In the actual process of collecting wind speed data, outliers may occasionally appear due to sensor abnormalities, data transmission errors, and other reasons. Outlier data should be removed, otherwise it will affect the accuracy of modeling. This paper analyzes the relationship between the mean and variance of wind speed samples to determine whether there are outliers, also known as the Pauta criterion. Assuming the wind speed data is $x(i), i = 1, 2, ..., N$, i is sample time, N is the length of the sample. Calculate the mean value of the wind speed sample in formula (1) and the variance of the wind speed sample in formula (2). Only when the new wind speed sample satisfies formula (3), it is considered reasonable data, otherwise it will be judged as an outlier. For excluded outlier data, cubic spline interpolation method is used to supplement missing data.

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x(i) \tag{1}$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(x(i)-\mu)^2}{N-1}} \tag{2}$$

$$\mu - 3\sigma < x(i) < \mu + 3\sigma \tag{3}$$

After processing according to the above principles, the change curves of the two wind speed data sets are shown in Fig. 1.

Table 2 shows the statistics of the maximum, minimum, median, mean and standard deviation values of the two data sets used. From Table 2, it can be seen that the features of the two data sets are not completely similar, which can fully verify the accuracy of the model.

Proposed prediction model

Short-term wind speed has uncertainty and complexity. In response to these characteristics of short-term wind speed, this study first uses clustering method for preliminary classification of wind speed. Then, before using the LSTM model for prediction, CNN is used to extract deep feature rules from wind speed data, and the IWOA is used to optimize the hyper-parameters of the LSTM model for predicting short-term wind speed. The flowchart of the proposed prediction model is shown in Fig. 2.

The corresponding implementation steps are described below.

Step 1 Preprocess the original short-term wind speed data to remove outliers and other unreasonable data. Generate training and testing sets. Usually, the training set accounts for 80% of all data, and the testing set accounts for 20% of all data. Due to the use of IWOA for model parameter optimization in this paper, no validation set is employed.

Step 2 The elbow method is used to perform K-means clustering on the training set, resulting in k data sets after clustering.

Step 3 CNN parameter initialization. Then, CNN is used to extract features from the k clustered data sets separately.

Step 4 After feature extraction, the data is modeled and predicted using LSTM. IWOA is used to optimize the hyper-parameters of LSTM. k optimal LSTM models are obtained.

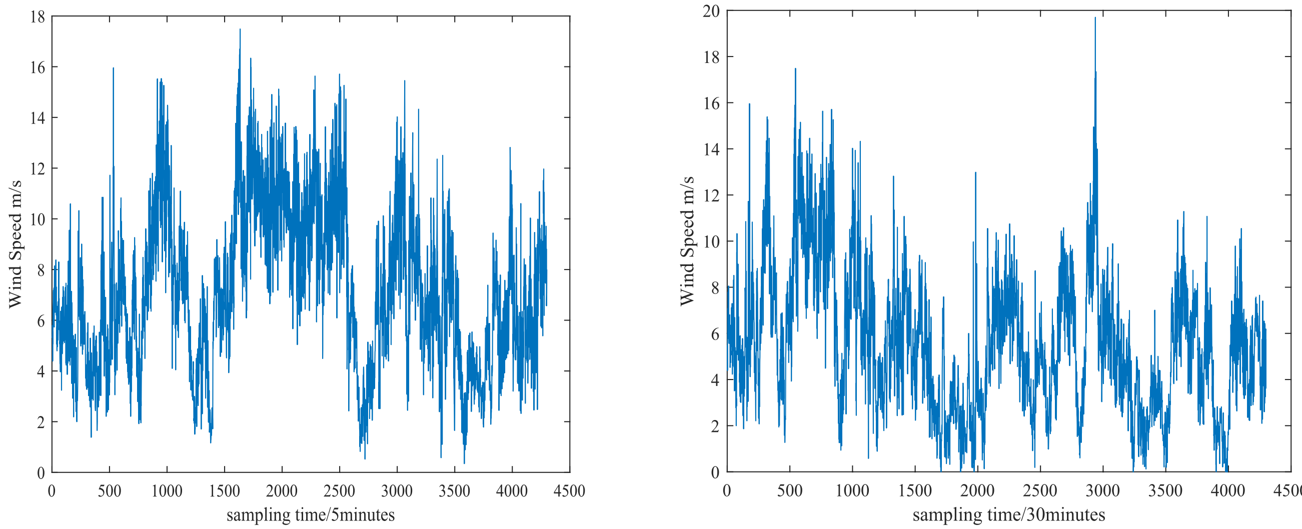


Fig. 1 The short-term wind speed data sets

Table 2 The statistical parameters of two data sets (m/s)

Data set	Maximum value	Minimum value	Mean value	Median	Standard deviation
Data set 1	17.488	0.344	7.221	6.983	3.061
Data set 2	19.701	0	5.584	6.983	3.074

Step 5 After the test set data is processed by K-means algorithm and CNN, it is input into the corresponding optimal LSTM to obtain the predicted values. Sort and reorganize the predicted values of each SLTM model to obtain the final prediction result.

The reason for the reordering of predicted values in Step 5 is the short-term wind speed is a time series. When training LSTM, it is necessary to store the relationship between input and output, that is, the time series number of the output value. For example, suppose the training set outputs a total of 5 values, and after clustering, 3 clusters are generated. The output numbers of cluster 1 are 1 and 3, cluster 2 is 4, and cluster 3 is 2 and 5. Therefore, when the clustering results of the test set are predicted by three LSTM models, the output values also need to be sorted according to the above sequence number to obtain the final prediction result.

Materials and methods

In this part, K-means algorithm, CNN, LSTM and IWOA are introduced.

K-means algorithm

The purpose of using the K-means algorithm is to cluster the raw short-term wind speed data. The clustered wind speed data is more regular, enhancing the stability of the wind speed data and facilitating the subsequent modeling process (Chen et al. 2022). The reason is that low data regularity can lead to significant errors in the modeling process. The K-means algorithm classifies data samples with similar features into the same cluster based on a certain degree of similarity between data, achieving clustering and partitioning of data.

Firstly, randomly select a point as the cluster center of the sample, then calculate the Euclidean distance between other samples and the cluster center, and classify them into the category closest to the center to form the initial classification. The process is as follows.

$$d(A,B) = \sqrt{\sum (a_t - b_t)^2} \quad t = 1,2,...,n \tag{4}$$

where a_t is the variable in sample A, and b_t is the variable in sample B.

Secondly, determine a standard function to evaluate the goodness of clustering metrics. The Davies-Bouldin

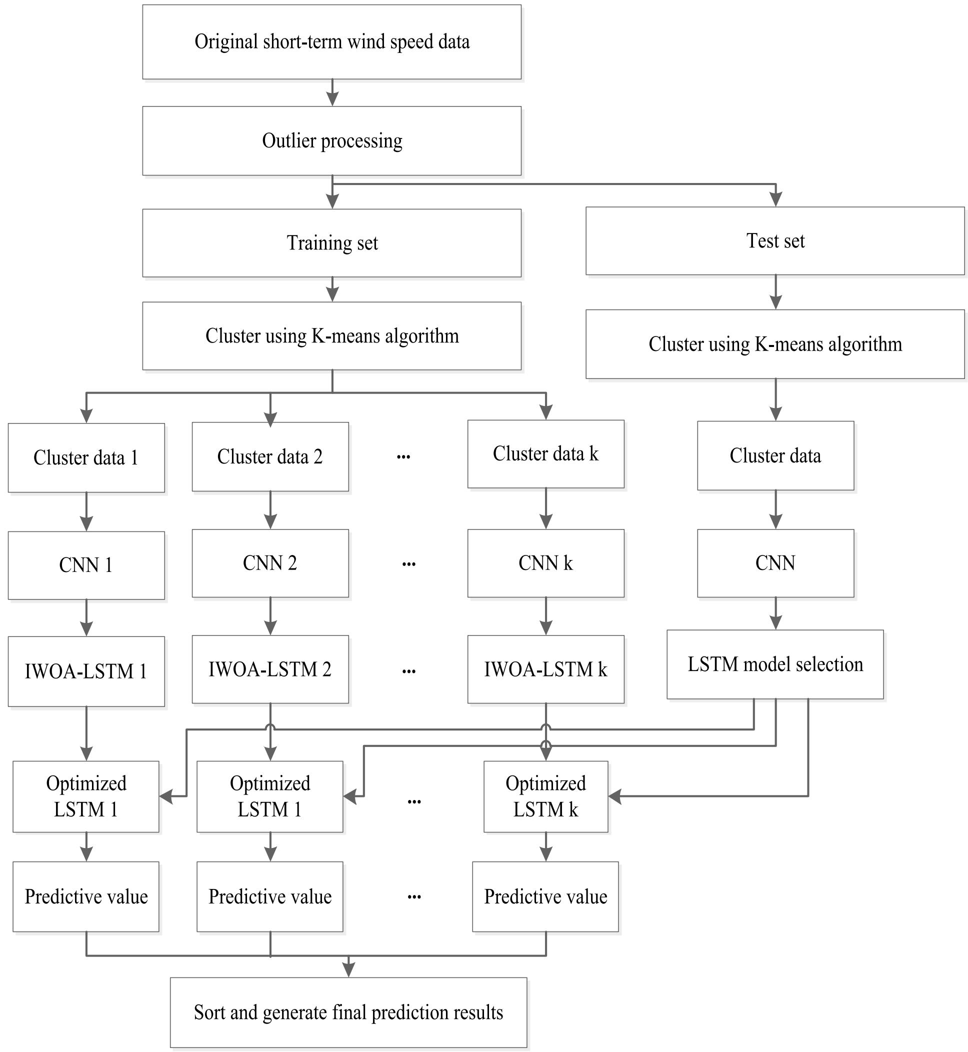


Fig. 2 The flowchart of the proposed prediction model

Index (DBI) can be used to measure the similarity between clusters.

$$\mathrm{DBI} = \frac{1}{K}\sum_{i=1}^{K}\max_{j\neq i}\left(\frac{S_i + S_j}{M_{ij}}\right) \tag{5}$$

where k is the number of cluster, S_i and S_j are the average distance of i and j samples and samples to the cluster center of the corresponding cluster, namely the average distance within the cluster. M_{ij} denotes the distance between the cluster i and the center j.

Finally, given an initial class, iterative algorithms are used to find a suitable center point to minimize the standard DBI function, that is, to minimize the similarity of each class as much as possible to obtain appropriate clustering results.

The K-means algorithm can reduce the impact of uncertainty in raw wind speed data, extract preliminary features of raw wind speed data, and form initial classifications with certain regularity. Usually, the elbow method is used to determine the number of clusters. The core metric of the elbow method is the sum of squared errors (SEE).

$$\mathrm{SSE} = \sum_{i=1}^{k}\sum_{u\in D_t}\left|u - r_t\right|^2 \tag{6}$$

where, D_t is the cluster t, u is the D_t centroid of sample point, r_t is the mean of all samples. SSE can demonstrate the accuracy of clustering effects.

With the increase of the number k of clustering, sample division will be more detailed, so the degree of clustering will be higher and higher, and SSE will be smaller and smaller. With the increase of the number of clustering, the sample division will be more detailed. And when k is smaller than the true number of clusters, the increase of k will greatly increase the aggregation degree of each cluster, so the SSE will decrease greatly. As k increases, the aggregation degree rapidly decreases, resulting in a sharp decline in SSE, which then tends to stabilize as k increases. This means that the relationship between SSE and k is a turning point, which corresponds to the true clustering value.

CNN model

CNN has excellent feature extraction capabilities and has been successfully applied to time series and fixed length signal data analysis. CNN typically includes convolutional layers, pooling layers, and fully connected layers (Rubio et al. 2024). Convolutional layers are crucial in extracting features and mapping from input data. Convolutional layers are used to extract features from wind speed sequences, pooling layers are used to reduce the dimensionality of wind speed sequences, compress features, and simplify complexity, and fully connected layers are used to convert data passed through pooling layers into one-dimensional data. The flowchart of CNN is shown in Fig. 3.

1D-CNN is mainly used for processing one-dimensional sequence data, such as audio, text, etc. Compared with traditional fully connected neural networks, 1D-CNN can better handle local relationships in sequential data (Han et al. 2022). Calculate the mapping matrix for different features.

$$X^{m+1} = \sigma\left(\sum W^{m}\cdot X^{m} + B^{m}\right) \tag{7}$$

where, X^{m} and X^{m+1} is the input of this layer and the next layer, W^{m} is the weight of this layer,B^{m} is biased, σ is the activation function. 1D-CNN cannot extract spatial features; therefore it cannot process data such as images. The short-term wind speed data used in this paper comes from the same wind turbine and does not include spatial features. It is a one-dimensional time series. If the wind speed data comes from multiple wind turbines, 2D-CNN needs to be used for prediction. For the research object of this paper, 1D-CNN is suitable.

LSTM model

LSTM is designed to eliminate the problem of long-term dependence of recurrent neural network (RNN). The LSTM model is composed of many memory model blocks, and each memory block contains three multiplicative control units: input gate, output gate and forget gate (Wen and Li 2023). The structure diagram of LSTM is shown in Fig. 4.

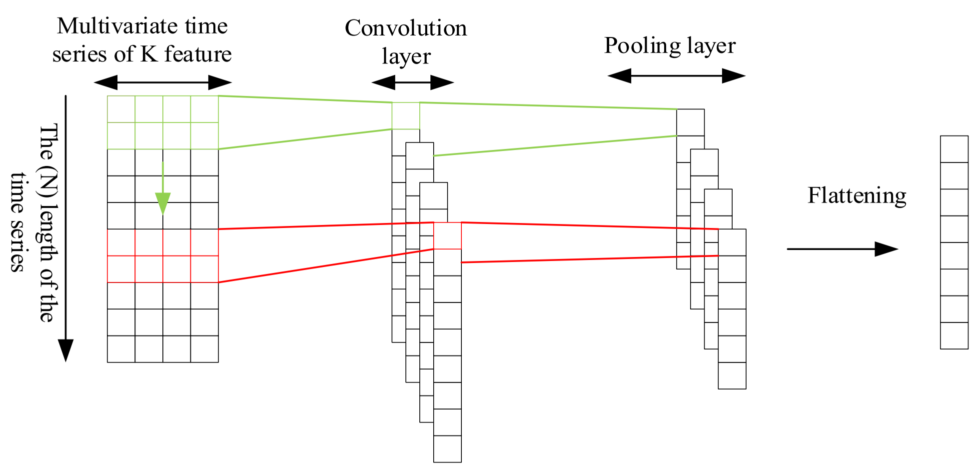


Fig. 3 The flowchart of CNN

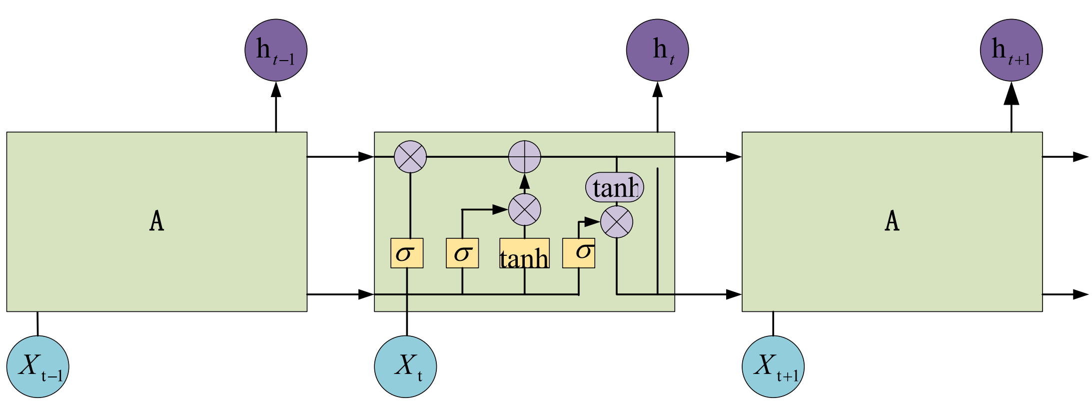


Fig. 4 The structure diagram of LSTM

The functions of each control unit are as follows.

Forget gate. It is used to determine the update of information, determined by an S-shaped layer called the forgetting threshold layer, which is represented by formula (8).

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{8}$$

where, W_f is the weight and b_f is the bias of the Sigmoid activation function of the forgetting gate.

Input gate. It is used to determine the input information represented by formula (9).

$$\begin{aligned} i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\ \tilde{C}_t &= \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \\ C_t &= f_t \times C_{t-1} + i_t \times \tilde{C}_t \end{aligned} \tag{9}$$

where, W_i is the weight and b_i is the bias of the Sigmoid activation function of the input gate, W_c is the weight and b_c is the bias of the input gate Tanh activation function, C_t is the newly obtained cell state information.

Output gate. Its function is to determine the value of the next hidden layer that can be represented by formula (10).

$$\begin{aligned} o_t &= \sigma(W_o [h_{t-1}, x_t] + b_o) \\ h_t &= o_t \times \tanh(C_t) \end{aligned} \tag{10}$$

where W_o is the weight and b_o is the bias of the Sigmoid activation function of the output gate.

For LSTM, the hyper-parameters include regularization coefficient, learning rate, and number of hidden layer nodes of LSTM have a significant impact on the performance of LSTM (Wang et al. 2024b). This paper uses the following proposed IWOA to optimize the hyper-parameters of LSTM.

IWOA

In recent years, the number of optimization algorithms inspired by nature and biology has been increasing. This includes ant colony algorithm, firefly algorithm, particle swarm optimization algorithm, etc. WOA is effective in terms of convergence speed and accuracy, achieving a good balance between global and local search. Therefore, this paper chooses the WOA algorithm.

WOA is an optimization algorithm that simulates the Whale predation process (Mirjalili and Lewis 2016). The Whale predation process is divided into three steps, namely three stages of WOA, which are encircling prey, bubble net attack and search predation respectively. Whale predation behavior means a group of Whales search for prey together, the population gradually obtains prey information through searching, and the Whales keep getting closer to the prey by encircling and spiraling the prey, and finally find the prey, that is, find the optimal solution (Uzer and Inan 2023). The detailed implementation process of WOA is as follows.

Step 1 Surrounding the prey stage.

When Whales hunt, they do not know the exact location of their prey, which requires Whales to communicate with each other and push the entire population towards the target prey. In WOA, it is assumed that the population size is N, the search space is d dimension, and the position of the i Whale in d dimension space can be expressed as $W_i = (w_i^1, w_i^2, ..., w_i^d)(i = 1, 2, ..., N)$. The mathematical model of enveloping predation is

$$\begin{aligned} E &= \left| F \cdot W_p(t) - W(t) \right| \\ W(t+1) &= W_p(t) - A \cdot E \end{aligned} \tag{11}$$

where t is the current number of iterations, $W(t)$ is the individual position vector, $W_p(t)$ is the position vector of prey (current optimal solution), A and F are the coefficient vectors respectively, and have,

$$\begin{aligned} A &= 2a \cdot g_1 - a \\ F &= 2 \cdot g_2 \end{aligned} \tag{12}$$

where g_1 and g_2 are random numbers of [0, 1] respectively. a is the control parameter, which linearly decreases from 2 to 0 as the number of iterations increases. In formula (13), max_iter is the maximum number of iterations.

$$a(t) = 2 - \frac{2t}{\max\_\text{iter}} \tag{13}$$

Step 2 Helical renewal stage.

The essence of the spiral update stage is for Whales to search for prey, with the prey as the center, constantly performing spiral movements to approach the prey. The following formula is a mathematical model.

$$W(t+1) = E \cdot e^{bl} \cdot \cos(2\pi l) + W_p(t) \tag{14}$$

where E is shown in formula (11), b is the constant used to define the logarithmic spiral shape, l is the random quantity of $[-1, 1]$. For realizing contraction encircling and spiral update synchronization, probability p is introduced to identify the position update mode. p belongs to [0, 1], and its mathematical model is shown in the following.

$$W(t+1) = \begin{cases} X_p(t) - A \cdot E & p < 0.5 \\ E \cdot e^{bl} \cdot \cos(2\pi l) + W_p(t) & p < 0.5 \end{cases} \tag{15}$$

Step 3 Prey stage.

The $|A|$ value controls whether Whales are in the hunting stage or searching for prey. When $|A| > 1$, Whales are unable to provide useful information to their prey, and they constantly attempt to search for clues about their prey in a random way. The mathematical model of their hunting is as follows.

$$E = \left|F \cdot W_{rand}(t) - W(t)\right|$$
$$W(t+1) = W_{rand}(t) - A \cdot E \tag{16}$$

where W_{rand} is the position vector of individual Whales randomly selected from the current population.

However, there is an imbalance between global search capability and local development capability in the implementation process of WOA. When parameter $|A| \geq 1$, the algorithm starts with the probability of 0.5 goes with the searches globally, and when $|A| < 1$, the algorithm develops local search. Due to the fact that the convergence factor cannot effectively adjust the global search ability and local development ability when linearly changing, this paper proposes a new nonlinear convergence factor with the following structure.

$$a = 2 - 3\sin(\mu \frac{t}{\text{max\_iter}} \pi + \phi) \tag{17}$$

where max_iter is the maximum number of iterations, t is the current number of iterations, μ and ϕ are the edge master correlation coefficient. Through experiments, select $\mu = 1/3, \phi = 0$.

This improved WOA is called IWOA. To validate the optimization performance of IWOA, the following benchmark functions in formulas (18) and (19) are selected for comparison with particle swarm optimization algorithm (PSO), Gray Wolf algorithm (GWO) and WOA.

Sphere function.

$$f_1(x) = \min \sum_{i=1}^{30} x_i^2, (-100 \leq x_i \leq 100) \tag{18}$$

Schwefe function.

$$F_2(x) = \min \sum_{i=1}^{30} [-x_i \sin(\sqrt{|x_i|})], -500 \leq x_i \leq 500 \tag{19}$$

For 4 optimization algorithms, the population size is uniformly set to 30, and the maximum number of iterations is 500. For PSO, $c1 = 1.5$, $c2 = 1.5$, $w = 0.9$. For WOA, $b = 1$, $a = [0, 2]$. For GWO, the convergence factor decreases linearly from 2 to 0, $r1 = [0, 1]$, $r2 = [0, 1]$. The algorithms are run 20 times and the average is taken as the result to eliminate the influence of randomness. The comparison results are shown in Figs. 5 and 6.

As shown in Figs. 5 and 6, IWOA has better convergence performance and speed than the other three optimization algorithms. Therefore, the IWO algorithm is chosen to optimize the hyper-parameters of the LSTM model.

IWOA optimized LSTM

The learning rate, regularization coefficient, and number of hidden layer nodes of LSTM can be regarded as a single Whale. The fitness function is selected as the Root Mean Square Error (RMSE) between the actual and predicted wind speed, and the LSTM model is trained using the training set. Through IWOA optimization iterations, the LSTM model is able to minimize RMSE and obtain optimal parameters.

Step 1 Firstly, initialize all parameters, including the LSTM model and the maximum number of iterations in

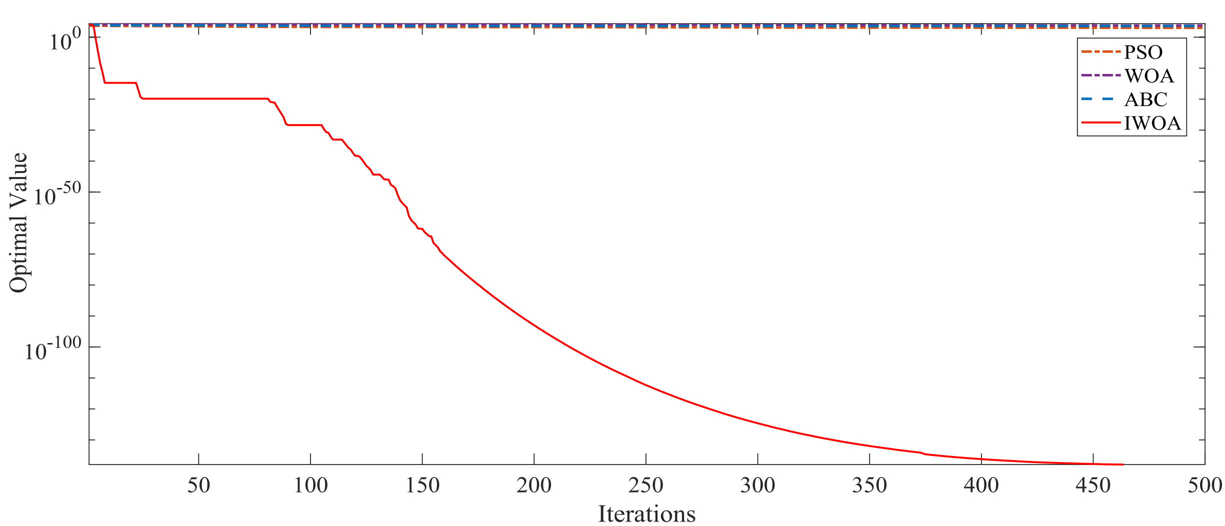


Fig. 5 Fitness comparison results of Sphere function

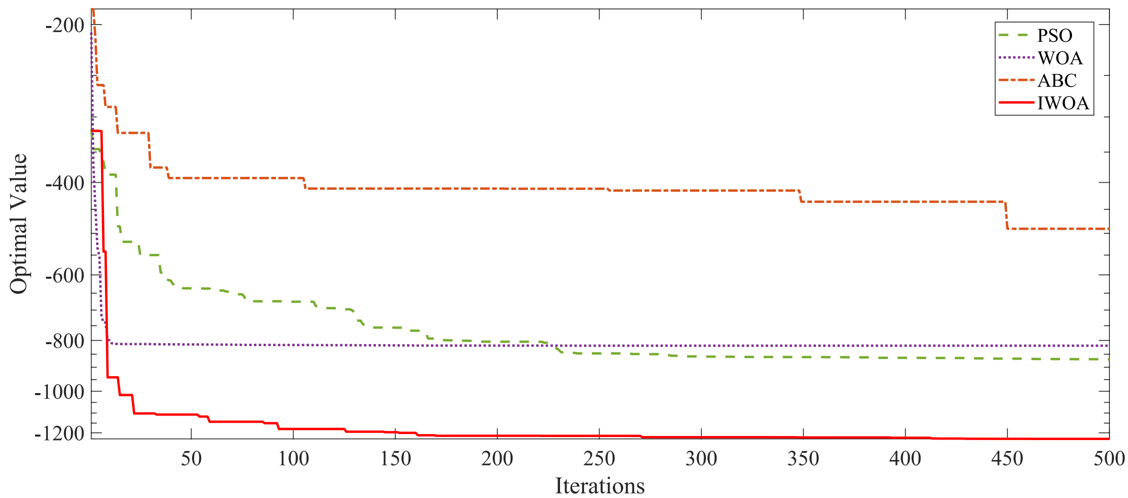


Fig. 6 Fitness comparison results of Schwefe function

IWOA, as well as population amount, Whale population dimension and location information, etc.

Step 2 The normalized interval of the training data set is [0, 1]. The normalization process is shown in formula (20). The advantage of normalization is that it can eliminate scale differences in data, accelerate model convergence, and improve model performance.

$$x_{norm} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{20}$$

where x is sample data, $x_{\max}$ is the maximum value of x, $x_{\min}$ is the minimum value of x, x_{norm} is the normalized value of x.

Step 3 By treating optimization parameters as individuals of IWOA and training and modeling LSTM, and the present iteration number is assumed to be t, let $t = 1$. Population initialization.where $l(i)$ is the actual value, $l(i)'$ is the predicted value, and N is the number of sample.

Step 4 Input the data into the LSTM model and use obtained learning rate, regularization coefficient, and number of hidden layer nodes to output predicted values for calculating the fitness function. The fitness function is defined as follows.

$$fitness = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(l(i) - l(i)')^2} \tag{21}$$

where $l(i)$ is the actual value, $l(i)'$ is the predicted value, and N is the number of sample.

Step 5 Obtain the current best solution of the population and consider it as the optimal hyper-parameter combination for LSTM.

Step 6 If the termination condition is met, the optimization algorithm is terminated and the parameters are updated to ultimately obtain the optimal parameters. Otherwise, $t = t + 1$, go back to Step 4 to continue the iteration

Experiment and results

Error evaluation indicators

In the process of short-term wind speed prediction, errors are inevitable, that is, the predicted wind speed is different from the actual wind speed. Common error evaluation indicators include: Mean Absolute Percentile Error (MAPE), R squared (R^2), Mean Absolute Error (MAE), RMSE, and Relative Root Mean Square Error (RRMSE) (Tian and Wei 2025; Sun and Tian 2024).

Generally, the purpose of using y_i is to represent the predictive value, and the purpose of using y_i' is to represent the actual value. The forecast absolute error is defined as follows.

$$e_i = y_i' - y_i \tag{22}$$

MAPE. It represents the percentage of prediction error to actual value. Therefore, the larger the MAPE value, the greater the difference between the predicted value and the actual value, that is, the worse the prediction effect. The definition of MAPE is as follows, where n represents the number of samples.

$$e_{\mathrm{MAPE}} = \frac{1}{n}\sum \frac{|e_i|}{y_i} \times 100\% \tag{23}$$

R^2. The meaning of r-squared value is an indicator indicating the degree of fitting, and its value is used to reflect the degree of fitting of the trend line between the predicted value and the actual value. The higher the fitting degree, the higher the reliability of the trend line.

$$e_{\mathrm{R}^2} = 1 - \frac{\sum_i (y_i' - y_i)^2}{\sum_i (\overline{y} - y_i)^2} \tag{24}$$

where $\overline{y}$ is the mean value of samples.

MAE. This indicator is defined as follows and is commonly used to evaluate the average margin of error in the forecast process.

$$e_{\mathrm{MAE}} = \frac{\sum |e_i|}{n} \tag{25}$$

RMSE. The definition of indicators is as follows, used to measure the degree of dispersion of errors.

$$e_{\mathrm{RMSE}} = \sqrt{\frac{\sum e_i^2}{n}} \tag{26}$$

RRMSE. It represents the relative difference between predicted and actual values, and is a dimensionless statistic that can be used to compare different variables. The smaller the value of RRMSE, the higher the prediction accuracy of the prediction model. Its definition is as follows.

$$e_{\mathrm{RRMSE}} = \sqrt{\frac{1}{n}\sum \left(\frac{e_i}{y_i}\right)^2} \tag{27}$$

The above error indicators constitute a classic error measurement method based on the average idea of point by point summation. Currently, almost all short-term wind speed predictions use a combination of one or more performance indicators to evaluate the prediction results (Wang and Tian 2024). Then the prediction model is modified and improved accordingly according to the error form.

Experimental results

Firstly, the K-means algorithm is used to process short-term wind speed data to obtain the number of clusters. For Data set 1, the clustering results are shown as Fig. 7. Similarly, the clustering results of Data set 2 are shown in Fig. 8.

The results of Figs. 7 and 8 indicate that as the number of clustering layers increases from small to large, the SSE of the clustering results rapidly decreases and gradually becomes stable. However, when k is 4, the SSE curve exhibits a stable characteristic. Therefore, the elbow point is set to 4, which means that for both data sets, the number of clusters is 4.

Figures 9 and 10 shows the clustering results of two short-term wind speed data sets. Table 3 shows the number of samples in the four clustered datasets of Data set 1 and Data set 2 after clustering.

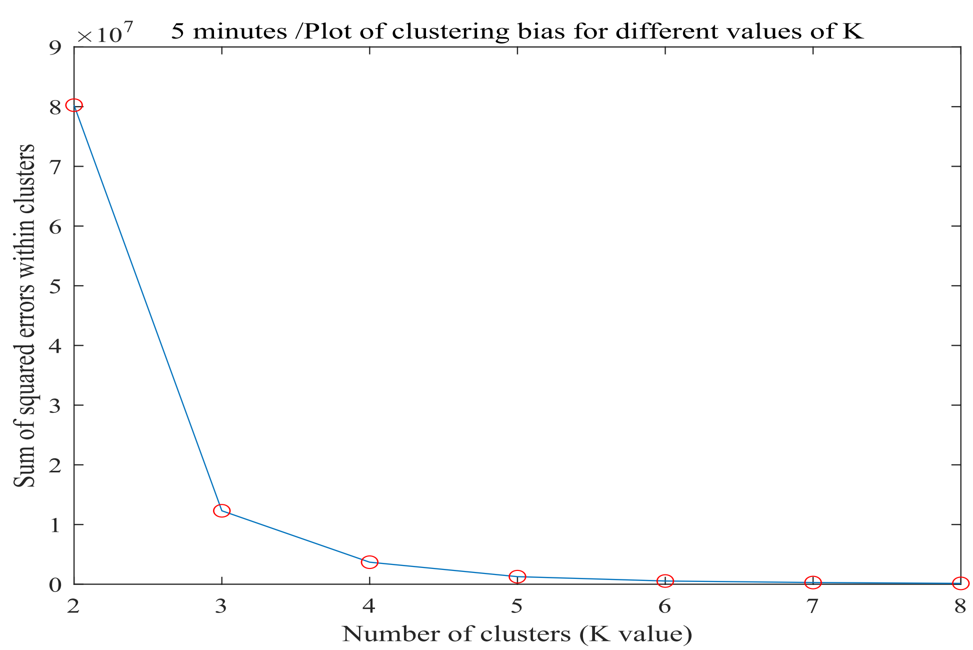


Fig. 7 The change curve of the number of clusters and SSE for Data set 1

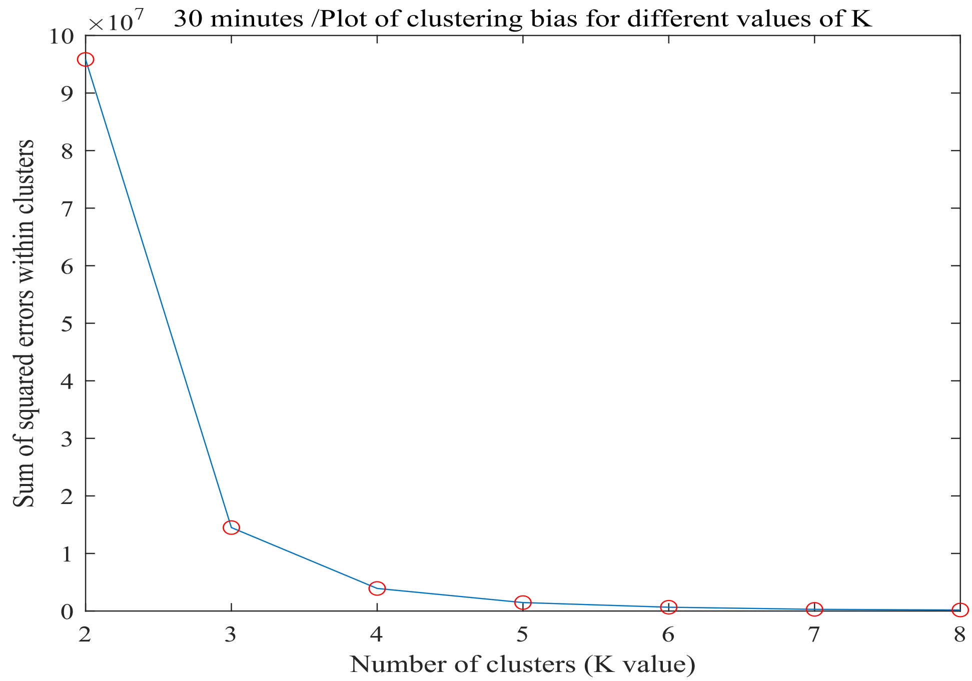


Fig. 8 The change curve of the number of clusters and SSE for Data set 2

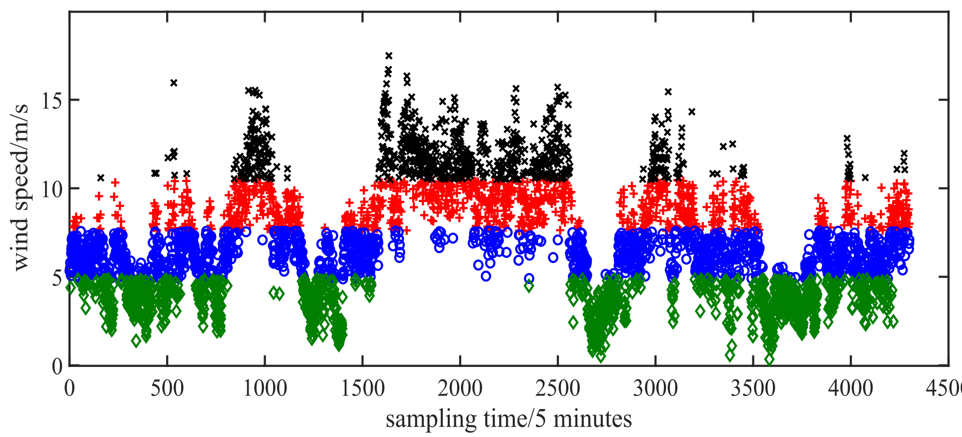


Fig. 9 The clustering results of Data set 1

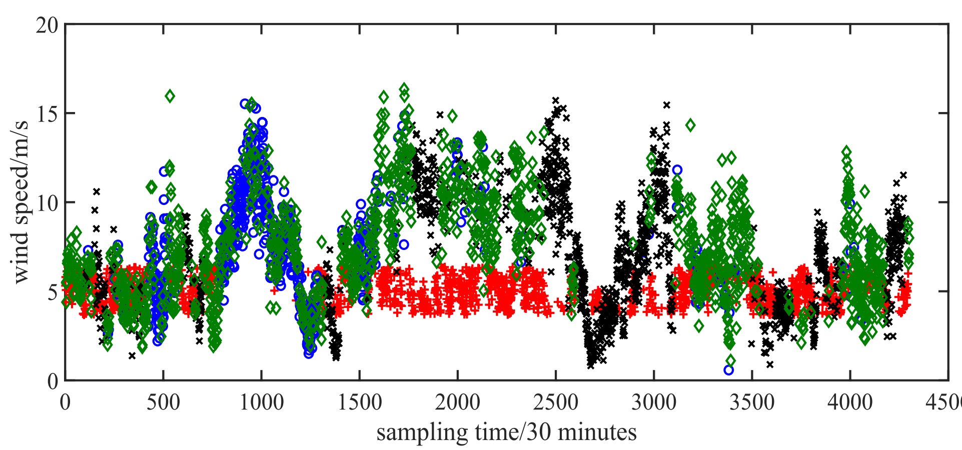


Fig. 10 The clustering results of Data set 2

Table 3 The number of samples in the four clustered datasets of Dataset 1 and Dataset 2

Cluster number	Dataset 1	Dataset 2
1	1105	1377
2	1354	558
3	732	1128
4	1109	1237

For Data set 1 and Data set 2, the first 80% is the training set and the last 20% is the testing set. The training set is input into the 1D-CNN. Before the data is inputting into the 1D-CNN, the data is first normalized, which can speed up the operation speed of the CNN. The 1D-CNN has two convolutional layer and two pooling layers. The convolution kernel dimension of its two convolution layers is 9*1 and 7*1 respectively, the two pooling layers are both 1*1, there is a fully connected layer, and the learning rate is 0.01. The activation function is chosen as Relu function, batch size is 64, stochastic gradient descent algorithm is used to optimize hyper-parameters. Such design can better extract data feature, it also speeds up the model.

After clustering, the wind speed data training set is processed by 1D-CNN to feature extraction and input into the IWOA optimized LSTM model for training model parameters. After the training is completed, the input test set is used to test the prediction accuracy of the model.

In this study, for IWOA, the maximum number of iterations is set to 20, the population size is set to 10. The regularization coefficient, learning rate, and number of hidden layer nodes of LSTM are used as optimization objects of IWOA. The ranges of parameters to be optimized for LSTM are: learning rate parameter range [0.001, 1], hidden number of nodes range [10, 300], and regularization coefficient range [0.001, 1]. After IWOA optimization, the hyper-parameters of LSTM corresponding to the four clustered datasets are shown in Table 4.

To verify the effectiveness of the proposed model, the following models include LSTM, K-means-LSTM, CNN-LSTM, RNN (Li et al. 2019a), and improved ELM (Tian et al. 2018) are selected for comparison and validation. The detailed parameters of these comparison models are given in Table 5.

Figures 11 and 12 respectively show the prediction results of the proposed model and the comparative models for two wind speed data sets. By comparing and analyzing the curves in two figures, it can be clearly seen that when the input data is Data set 1 and Data set 2, the predicted result curve of the proposed model is highly consistent with the curve of the original data. Not only does it maintain consistency in overall trends, but it also closely follows changes in the original short-term wind speed data at many data points. Although there are some differences at individual peak vertices, overall, these differences are not significant and do not affect the overall prediction performance of the model.

Table 6 shows the error performance indicators of Data set 1 predicted by each model. Similarly, Table 7 presents the relevant results for Data set 2. The training time of the models is also very important, and the training time of these models is also given in the two tables. The configuration information of the simulation computer is that the CPU is i5-10210U (1.6GHz with quad core); The memory is 8GB; The operating system is Windows 10; The simulation software is Matlab 2018b.

By comparing the performance indicators in Tables 6 and 7 in detail, it can be clearly seen that the proposed model exhibits higher prediction accuracy and better stability in both prediction scenarios. This further proves its superior performance in short-term wind speed prediction. Due to the consideration of many complex factors and the integration of advanced optimization algorithms during the construction of the proposed model, these characteristics require more computing resources and time during the training process. Although this increases the time cost of training to some extent, it also endows the model with stronger learning ability and higher prediction accuracy. Meanwhile, the training time of the model is less than the sampling time of the wind speed, which can meet the practical application requirements.

Table 4 The hyper-parameters of LSTM corresponding to the four clustered datasets

Cluster number	The hyper-parameters of LSTM (Data set 1)	The hyper-parameters of LSTM (Data set 2)
1	The learning rate is 0.586, the number of hidden layer nodes is 46, and the regularization coefficient is 0.362	The learning rate is 0.032, the number of hidden layer nodes is 49, and the regularization coefficient is 0.014
2	The learning rate is 0.027, the number of hidden layer nodes is 52, and the regularization coefficient is 0.251	The learning rate is 0.066, the number of hidden layer nodes is 54, and the regularization coefficient is 0.283
3	The learning rate is 0.164, the number of hidden layer nodes is 47, and the regularization coefficient is 0.301	The learning rate is 0.063, the number of hidden layer nodes is 56, and the regularization coefficient is 0.118
4	The learning rate is 0.093, the number of hidden layer nodes is 59, and the regularization coefficient is 0.196	The learning rate is 0.074, the number of hidden layer nodes is 49, and the regularization coefficient is 0.274

Table 5 The parameters of the comparison models

Model	Data set 1	Data set 2
LSTM	The number of hidden layer nodes is 200, the number of iterations is 600, the learning rate is 0.01, the regularization rate is 0.001, and the batch size is 64	The number of hidden layer nodes is 200, the number of iterations is 600, the learning rate is 0.01, the regularization rate is 0.001, and the batch size is 64
K-means-LSTM	The number of clusters is 4. LSTM: the number of hidden layer nodes is 200, the number of iterations is 600, the learning rate is 0.01, the regularization rate is 0.001, and the batch size is 64	The number of clusters is 4. LSTM: the number of hidden layer nodes is 200, the number of iterations is 600, the learning rate is 0.01, the regularization rate is 0.001, and the batch size is 64
CNN-LSTM	CNN: two convolutional layers, with convolutional kernels of 9*1 and 7*1, a learning rate of 0.01, batch size of 64, and a Relu activation function. LSTM: The number of hidden layer nodes is 100, the number of iterations is 300, the learning rate is 0.1, the regularization rate is 0.001, and the batch size is 64	CNN: two convolutional layers, with convolutional kernels of 9*1 and 7*1, a learning rate of 0.01, batch size of 64, and a Relu activation function. LSTM: The number of hidden layer nodes is 100, the number of iterations is 300, the learning rate is 0.1, the regularization rate is 0.001, and the batch size is 64
RNN	Input layer is 2, LSTM layer is 200, fully connected layer 1 is 50, drop out layer is 0.5, fully connection layer 2 is 2, regression layer is 1	Input layer is 2, LSTM layer is 200, fully connected layer 1 is 50, drop out layer is 0.5, fully connection layer 2 is 2, regression layer is 1
Improved ELM	m is 30, L is 100, μ is 0.9, λ is 1	m is 30, L is 100, μ is 0.9, λ is 1

The following are histograms of prediction errors for each model of two types of wind speed data. Figure 13 shows the histogram of prediction errors for 5-min dataset of each model, and Fig. 14 shows the histogram of expected errors for 30-min dataset of each model. From the figures, it can be seen that the designed model has a more uniform pattern, smaller errors, and better predictions.

For the two wind speed data sets, the Taylor plots shown in Figs. 15 and 16 are used to clarify the differences in prediction performance of each prediction model. In the figure, the standard deviation, root mean square deviation (RMSD), and correlation coefficient between each prediction model and observed (actual) wind speed values are clearly displayed. The horizontal and vertical axes in the figure represent the standard deviation, and the fan-shaped curve represents the correlation coefficient. The dashed line represents RMSD. Point A represents the observed (actual) value of wind speed, while other points represent the standard deviation, RMSD, and correlation coefficient of the proposed prediction model and comparison model. From the figure, it can be seen that point B (proposed model) is closer to point A (actual value). Therefore, the correlation coefficient of the proposed prediction model is greater than that of the comparison model, which means that the predicted values of the proposed prediction model are more consistent with the observed (actual) values.

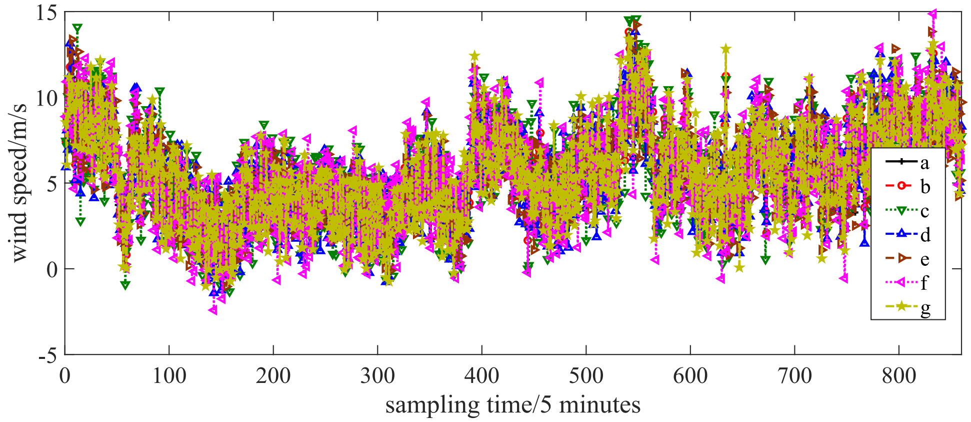
In the following Figs. 17 and 18, the radar chart displays the standard deviation of error for each prediction model. The closer the line in the figure is to the center of the radar, the smaller the standard deviation of the corresponding model error, indicating that the prediction accuracy of the model is higher. For the 5-min and 30-min wind speed datasets, through the observation of the two figures, it can be seen that our model shows lower errors, which proves that this model has better short-term wind speed fitting ability than all other comparison models.

Discussions

The study of two short-term wind speed data sets fully demonstrates that the proposed model is superior to other models (except for training time). The following is some analysis and discussion of the results.

1. By using the K-means algorithm for clustering, short-term wind speed data with similar features will be grouped into one cluster. Data samples with similar features can reduce the requirements for model performance and improve prediction accuracy. The processing of CNN will further make the feature information more prominent and regular. The LSTM optimized by IWOA has the optimal hyper-parameters to ensure prediction

Fig. 11 The prediction results of the proposed model and the comparative models (Data set 1) (**a** actual, **b** proposed model, **c** LSTM, **d** K-means-LSTM, **e** CNN-LSTM, **f** RNN, **g** Improved ELM)



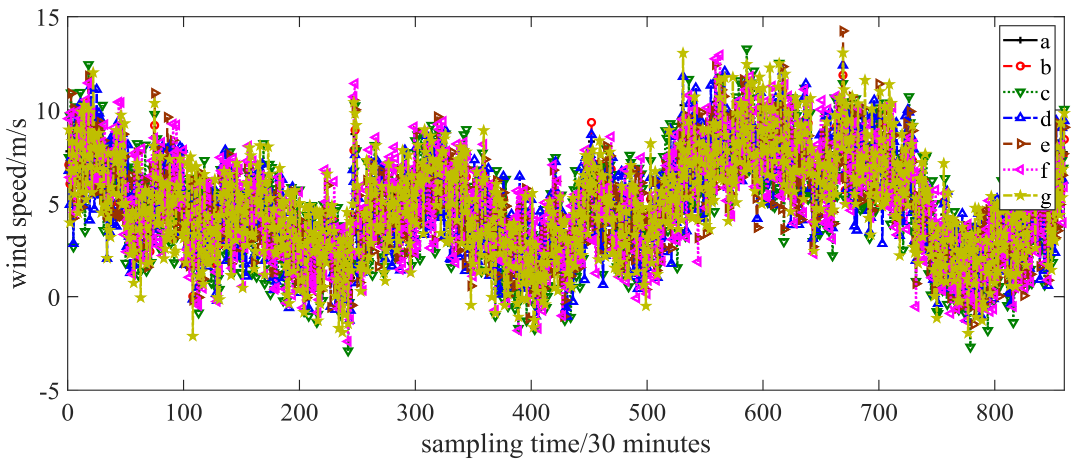


Fig. 12 The prediction results of the proposed model and the comparative models (Data set 2) (**a** actual, **b** proposed model, **c** LSTM, **d** K-means-LSTM, **e** CNN-LSTM, **f** RNN, **g** Improved ELM)

Table 6 Error indicators of each model for Data set 1

The model	MAPE(%)	MAE(m/s)	RMSE(m/s)	R^2	RRMSE	Run time (s)
LSTM	32.7890	1.5195	1.7423	0.6345	0.4182	15.435
K-means-LSTM	30.8068	1.4078	1.6185	0.6516	0.4253	10.705
CNN-LSTM	25.7726	1.1730	1.3636	0.7371	0.3724	105.432
RNN	36.8419	1.6216	1.8504	0.6105	0.5359	14.325
Improved ELM	31.5158	1.4414	1.6843	0.6376	0.4165	10.485
Our model	11.2075	0.5018	0.5777	0.9393	0.1601	221.604

Table 7 Error indicators of each model for Data set 2

The model	MAPE(m/s)	MAE(m/s)	RMSE(m/s)	R^2	RRMSE	Run time (s)
LSTM	46.9017	1.6705	1.9062	06873	0.7812	17.034
K-means-LSTM	36.9097	1.3933	1.6254	0.7563	0.5162	11.222
CNN-LSTM	34.2919	1.2779	1.4816	0.7920	0.4792	102.166
RNN	42.2605	1.5374	1.8059	0.6930	0.6903	12.528
Improved ELM	43.4552	1.5593	1.8120	0.6938	0.6942	9.403
Our model	15.0919	0.5513	0.6300	0.9256	0.2236	218.174

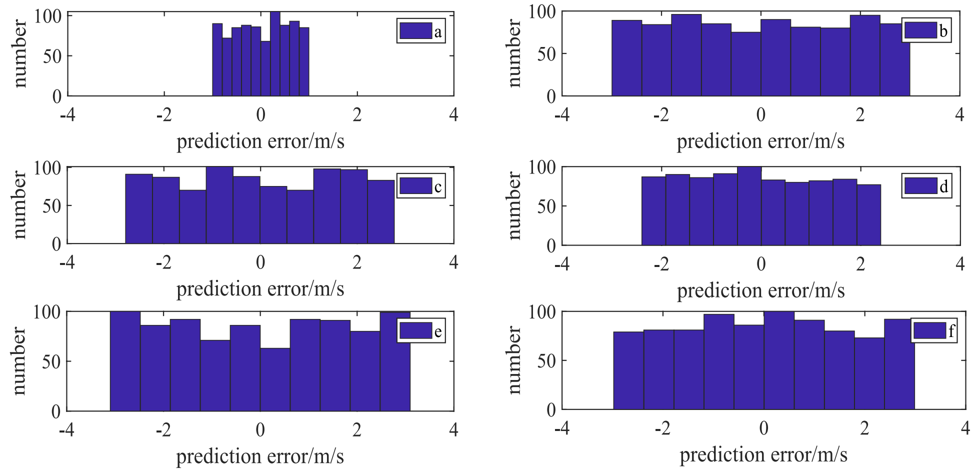


Fig. 13 Histogram of prediction error of each model in 5-min data set (a: proposed model, b: LSTM, c: K-means-LSTM, d: CNN-LSTM, e: RNN, f: Improved ELM)

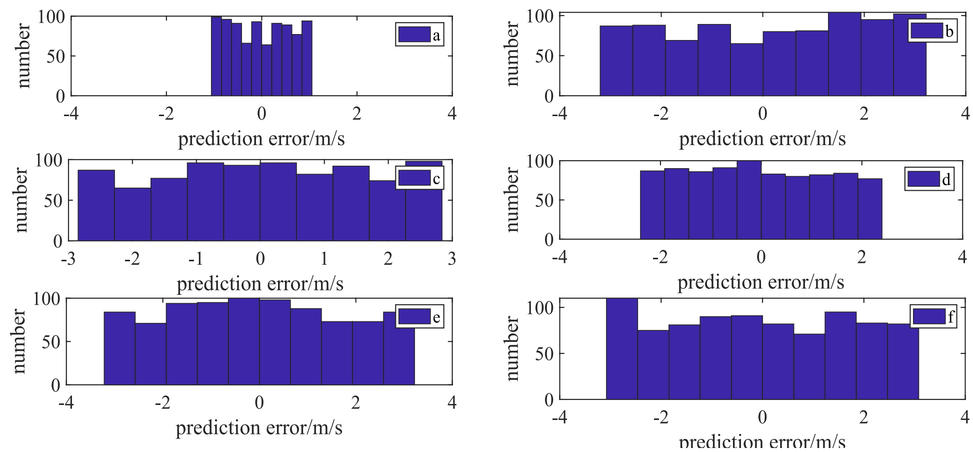


Fig. 14 Histogram of prediction error of each model in 30-min data set (**a** proposed model, **b** LSTM, **c** K-means-LSTM, **d** CNN-LSTM, **e** RNN, **f** Improved ELM)

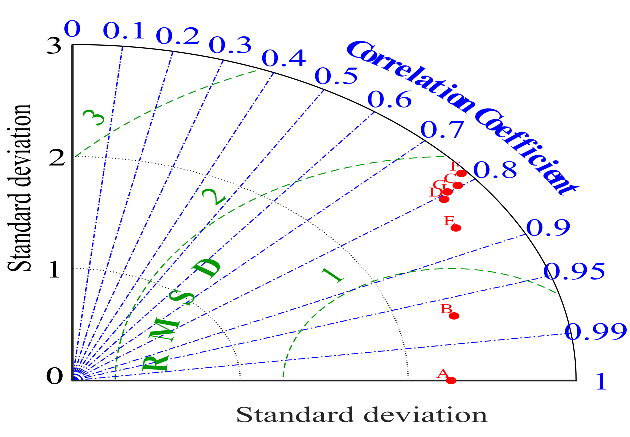


Fig. 15 The Taylor plots of each model (Data set 1) (**A** actual, **B** proposed model, **C** LSTM, **D** K-means-LSTM, **E** CNN-LSTM, **F** RNN, **G** Improved ELM)

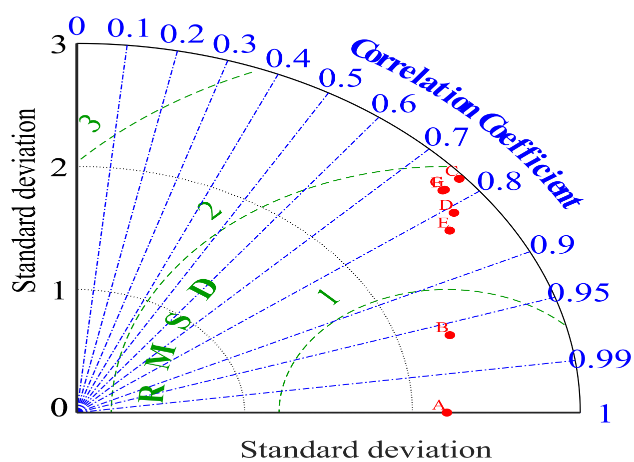


Fig. 16 The Taylor plots of each model (Data set 2) (**A** actual, **B** proposed model, **C** LSTM, **D** K-means-LSTM, **E** CNN-LSTM, **F** RNN, **G** Improved ELM)

accuracy. These are the reasons why the proposed model outperforms other models.

2. According to the result analysis in Figs. 11 and 12, the predicted values of the short-term wind speed model established by the two sets of data are basically consistent with the trend of actual short-term wind speed changes.

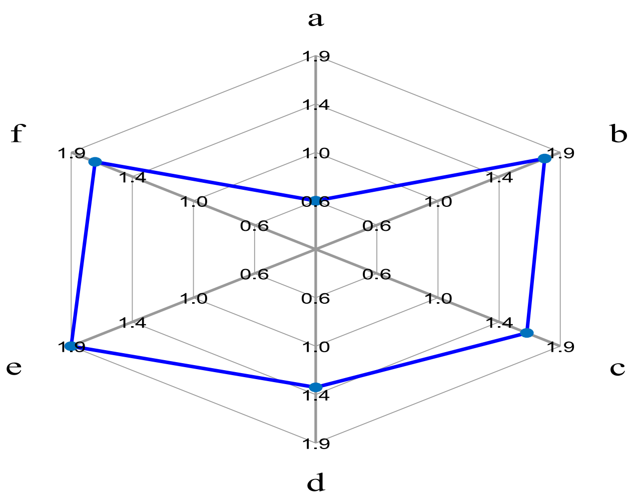


Fig. 17 The radar chart of each model (Data set 1) (**a** proposed model, **b** LSTM, **c** K-means-LSTM, **d** CNN-LSTM, **e** RNN, **f** Improved ELM)

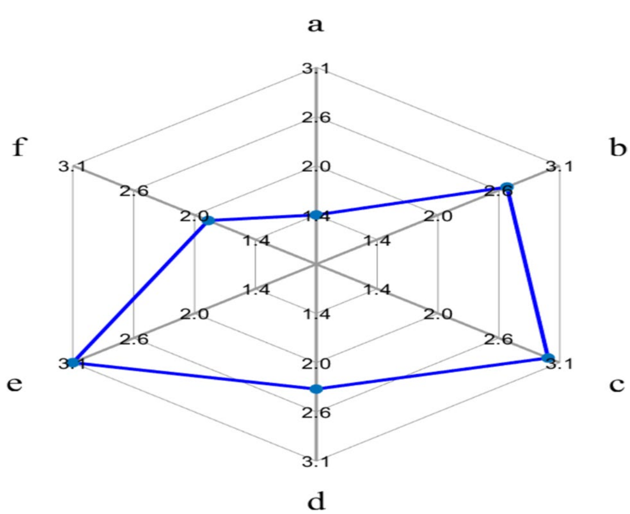


Fig. 18 The radar chart of each model (Data set 2) (**a** proposed model, **b** LSTM, **c** K-means-LSTM, **d** CNN-LSTM, **e** RNN, **f** Improved ELM)

3. As can be seen from the histograms in Figs. 13 and 14, the prediction error of the proposed prediction model is smaller than that of other models, and the uniformity is better than that of other models.
4. According to the data comparison results in Tables 6 and 7, MAPE, MAE, RMSE and RRMSE of the established prediction model are smaller than those of the comparison models. Meanwhile, R^2 of the proposed model is greater than other models. R^2 represents the degree of fit between predicted data and actual data, the proposed model has the best fit to short-term wind speed data. Therefore, the proposed prediction model has better prediction effect.
5. The Taylor plots in Figs. 15 and 16 further demonstrate the superiority of the proposed model from the perspective of data correlation, as well as the radar plots in Figs. 17 and 18 from the perspective of standard deviation.
6. Based on the relevant results, it can be concluded that the proposed prediction model demonstrated better performance on the 5-min Data set 1 than 30-min Data set 2. The main reason for the performance difference is that the complexity characteristics of the two data sets are different. As shown in Table 2, the maximum, minimum, standard deviation, and other data differences in the 30 min data set are greater. Furthermore, the wind speed variation in Fig. 1 also indicates that compared to the 5-min dataset, the 30-min wind speed dataset exhibits stronger nonlinear and random characteristics, and the data difference between the sampling points before and after is greater. This puts higher demands on the performance of the prediction model, resulting in the performance of the prediction model on the 30-min data set being inferior to that on the 5-min dataset. The corresponding comparison results show that for 5-min or 30-min datasets, the proposed prediction model exhibits better performance than the compared models. Therefore, the proposed prediction model is still effective.

Conclusions

This paper proposes a short-term wind speed prediction model based on LSTM with feature extraction. K-means clustering algorithm is used to cluster short-term wind speed, and then 1D-CNN is used for feature extraction to extract the law of wind speed. Finally, the wind speed data extracted by the feature is input into the LSTM model optimized by IWOA to obtain the final prediction result. The proposed model with a sampling period of 5-min and 30-min are taken as the sample set for case study. By comparing with other models, we can conclude that the proposed prediction model is feasible. The research results can provide certain assistance for wind speed forecasting in practical engineering.

Extracting features from clustering data using CNN may result in the loss of some features such as intermittency and randomness in wind speed data, which affects the robustness of the model and is also a limitation of CNN. For the information loss caused by CNN feature extraction, attention mechanisms such as squeeze-and-excitation attention mechanism can be introduced into CNN to increase the robustness of the model.

Author contributions Zhongda Tian: Conceptualization, Methodology, Software, Validation, Writing, and Funding acquisition. Xiyan Yu: Software, Validation, Writing. Guokui Feng: Software, Validation, Writing.

Xiyan Yu: Software, Validation, Writing.

Guokui Feng: Software, Validation, Writing.

Funding This paper is supported by the Natural Science Foundation of Liaoning Province of China (No. 2020-MS-210), the Science Research Project of Liaoning Education Department (No. LJKZ0143), and the Open Project of State Key Laboratory of Synthetical Automation for Process Industries (2023-kfkt-01).

Data availability No datasets were generated or analysed during the current study.

Declarations

Ethical approval Not applicable.

Consent to participate Not applicable.

Consent to publish The manuscript was vetted and approved for publication by all authors.

Competing interests The authors declare no competing interests.

References

Aasim A, Singh SN, Mohapatra A (2019) Repeated wavelet transform based ARIMA model for very short-term wind speed forecasting. Renew Energy 136:758–768

Chen GG, Tang BR, Zeng XJ, Zhou P, Kang P, Long HY (2022) Short-term wind speed forecasting based on long short-term memory and improved BP neural network. Int J Electr Power Energy Syst 134:107365

Cui Y, Yan S, Zhang HQ, Huang SY (2019) Ultra-short-term prediction of wind power based on chaos theory and ABC optimized RBF neural network. In: Proceedings of 2019 IEEE 3rd International Electrical and Energy Conference (CIEEC 2019). IEEE, pp 1422–1427

Duan JK, Chang MH, Chen XY, Wang WP, Zuo HC, Bai YL, Chen BL (2022) A combined short-term wind speed forecasting model based on CNN–RNN and linear regression optimization considering error. Renew Energy 200:788–808

Han Y, Mi LH, Shen L, Liu YC, Li K, Xu GJ (2022) A short-term wind speed prediction method utilizing novel hybrid deep learning algorithms to correct numerical weather forecasting. Appl Energy 312:118777

Hoolohan V, Tomlin AS, Cockerill T (2018) Improved near surface wind speed predictions using Gaussian process regression combined with numerical weather predictions and observed meteorological data. Renew Energy 126:1043–1054

Jaeseok C, Do DN, Lee YC (2016) Hourly average wind speed simulation and forecast based on ARMA model in Jeju Island, Korea. J Electr Eng Technol 11(6):1548–1555

Li F, Ren GR, Lee J (2019a) Multi-step wind speed prediction based on turbulence intensity and hybrid deep neural networks. Energy Convers Manage 186:306–322

Li YN, Yang P, Wang HJ (2019b) Short-term wind speed forecasting based on improved ant colony algorithm for LSSVM. Clust Comput 22:11575–11581

Li LL, Zhao X, Tseng ML, Tan RR (2020) Short-term wind power forecasting based on support vector machine with improved dragonfly algorithm. J Clean Prod 242:118447

Mirjalili S, Lewis A (2016) The Whale optimization algorithm. Adv Eng Softw 95:51–67

Rubio JD, Garcia D, Rosas FJ, Hernandez MA, Pacheco J, Zacarias A (2024) Engineering Applications of Artificial Intelligence 132:107998

Sun XL, Tian ZD (2024) A novel air quality index prediction model based on variational mode decomposition and SARIMA-GA-TCN. Process Saf Environ Prot 184:961–992

Tan L, Han J, Zhang HT (2020) Ultra-short-term wind power prediction by salp swarm algorithm-based optimizing extreme learning machine. IEEE Access 8:44470–44484

Tian ZD, Wei DL (2025) A hybrid wind speed forecasting model with two-stage data processing based on adaptive neuro-fuzzy inference systems and deep learning algorithms. Earth Sci Inf 18(1):140

Tian ZD, Ren Y, Wang G (2018) Short-term wind power prediction based on empirical mode decomposition and improved extreme learning machine. J Electr Eng Technol 13(5):1841–1851

Uzer MS, Inan O (2023) Application of improved hybrid whale optimization algorithm to optimization problems. Neural Comput Appl 35(17):12433–12451

Valdivia-Bautista SM, Dominguez-Navarro JA, Perez-Cisneros M, Vega-Gomez GJ, Castillo-Tellez B (2023) Artificial intelligence in wind speed forecasting: a review. Energies 16(5):2457

Wang MS, Tian ZD (2024) Ultra-short-term wind speed prediction based on empirical wavelet transform and combined model. Earth Sci Inf 17(1):539–560

Wang X, Wu QL, Tan GR, Qian DL, Zhou K (2023) A forecast model of short-term wind speed based on the attention mechanism and long short-term memory. Multimedia Tools Appl 83(15):45603–45623

Wang PJ, Long QL, Zhang H, Chen X, Yu R, Guo FQ (2024a) Forecasting and multilevel early warning of wind speed using an adaptive kernel estimator and optimized gated recurrent units. Mathematics 12(16):2581–2581

Wang QQ, Zheng MZ, Yang K, Shang CX, Luo Y (2024b) Research and implementation of fault data recovery method for dry-type transformer temperature control sensor based on ISSA-LSTM algorithm. Measurement 228:114333

Wen XY, Li WB (2023) Time series prediction based on LSTM-Attention-LSTM model. IEEE Access 11:48322–48331

Yang QL, Deng CH, Chang XQ (2022) Ultra-short-term/short-term wind speed prediction based on improved singular spectrum analysis. Renew Energy 184:36–44

Yu CQ, Yan GX, Yu CM, Liu XW, Mi XW (2024) MRIformer: A multi-resolution interactive transformer for wind speed multi-step prediction. Inf Sci 661:120150

Zhang ZY, Wang J, Wei D, Luo TR, Xia YR (2023) A novel ensemble system for short-term wind speed forecasting based on two-stage attention-based recurrent neural network. Renew Energy 204(6):11–23

Zhang GW, Zhang Y, Wang H, Liu D, Cheng RK, Yang D (2024) Short-term wind speed forecasting based on adaptive secondary decomposition and robust temporal convolutional network. Energy 288:129618

Zucatelli PJ, Nascimento EGS, Aylas GYR, Souza NBP, Kitagawa YKL, Santos AAB, Arce AMG, Moreira DM (2019a) Short-term wind speed forecasting in Uruguay using computational intelligence. Heliyon 5(5):e01664

Zucatelli PJ, Nascimento EGS, Arce AMG, Moreira DM (2019b) Short-range wind speed predictions in subtropical region using artificial intelligence. WMSCI 2019 - 23rd World Multi-conference on systemics. Cybern Inform 1:104–109.

Zucatelli PJ, Nascimento EGS, Santos AAB, Arce AMG, Moreira DM (2020) Nowcasting prediction of wind speed using computational intelligence and wavelet in Brazil. Int J Comput Methods Eng Sci Mech 21(6):343–369

Zucatelli PJ, Nascimento EGS, Santos AAB, Arce AMG, Moreira DM (2021) An investigation on deep learning and wavelet transform to nowcast wind power and wind power ramp: A case study in Brazil and Uruguay. Energy 230:120842