Wind Power Short-Term Forecasting Based on LSTM Neural Network With Dragonfly Algorithm

Liu Hui, Chen Dihuang*, Lin Fang, Wan Zhouli
Hubei Key Laboratory for High-efficiency Utilization of Solar Energy and Operation Control of Energy Storage System, Hubei University of Technology, Wuhan, 430068, P. R. China
*Corresponding author’s e-mail: 1414220229@mail.wtu.edu.cn

Abstract. The volatility and randomness of wind energy limit its large-scale usage in power systems. Accurate short-term wind power prediction can provide effective criteria for wind energy parallel in the grid and provide favorable conditions for the commercial utilization of wind energy. Therefore, the paper proposes a short-term wind power prediction model based on the dragonfly algorithm optimize long-term and short-term neural networks. Firstly, the model preprocesses the collected data and divides the data into a training set and a testing set. Then, the DA used the training set to optimize the relevant hyperparameters in the long and short-term memory neural network. Finally, the DA-LSTM prediction model constructed with excellent hyperparameters will use the test set to obtain the prediction results. The simulation results of the examples show that, compared with the GWO-BP, ELM, and LSTM models, the DA-LSTM model can effectively use time series data for short-term forecasting of wind power and has higher prediction accuracy.

1. Introduction
With the increasing awareness of green energy consumption and reducing environmental pollution on the global scale, renewable energy has gradually become an increasingly important part of the energy structure of the countries all over the world. Wind energy, as the energy source with the largest reserves, the smallest cost, the most mature technology, and the least environmental impact among the renewable energy sources, has become the fastest-growing emerging renewable energy source in recent years [1]. Although wind energy grid-connected has many benefits, due to the volatility and randomness of wind energy, the power generation of wind turbines is not artificially controllable, which leads to uncertainties in the power output characteristics of wind farms [2]. Therefore, when small-capacity wind turbines are connected to the grid, a series of power quality problems such as local harmonic pollution and voltage fluctuations will occur. When large-capacity wind turbines are connected to the grid, they will put tremendous pressure on the stability of the grid voltage [3]. In order to alleviate this impact, the power system needs to reserve peak-shaving power plants in advance for the grid connection of wind energy, which will lead to the increase of power generation cost and challenge the operation of power grid [4]. Thus, timely and accurate forecasting of wind energy can not only reduce the impact of wind power integration on the stable operation of the power system, but also help power related departments to formulate a reasonable dispatch plan. It can also reduce the reserve capacity of the grid, thereby reducing the commercial cost of wind power [5]. In addition, accurate forecasting of wind energy can provide convenience for arranging the maintenance of wind
farms, and it will also provide favorable conditions for the integration of wind energy into the energy market [6].

For a long time, many scholars at home and abroad have carried out in-depth research on short-term wind energy forecasting and proposed a large number of forecasting methods in order to solve the related problems of maintaining the stability of the power system when wind energy is connected to the grid. Literature [7] added the Yule-Walker equation on the basis of the traditional ARIMA algorithm to determine the autoregressive parameters of the model, and used the relationship between the serial autocovariance and the moving average parameters to determine the moving average parameters, and at the same time introduced the link. The forecast results are revised in the link. Literature [8] proposed a GA-SVM-based forecasting method, using genetic algorithms to optimize the relevant parameters of support vector machines and constructing a GA-SVM ultra-short-term wind power forecasting model. Literature [9] uses lifting wavelet transform to decompose the original data into low-frequency sequence and high-frequency sequence, and then uses an improved population competition algorithm to construct the LWT-IPCA-LSSVM prediction model for the parameters obtained by LSSVM optimization. Although these traditional forecasting methods have high accuracy in application scenarios with small data scales, as the energy structure of the power system has become more and more complex in recent years, the scale of power data has become larger and larger, resulting in the accuracy of these models cannot meet the needs of electricity production. Literature [10] proposed a wind power generation power prediction method based on long and short-term memory neural networks. Although this method has good prediction accuracy when applied to the prediction of time series with large data scale, the parameters of the model are troublesome to determine. And if it is directly applied to other actual forecasting problems, it may not show better forecasting performance.

Based on this, the paper proposes a short-term wind power forecasting method based on DA-LSTM neural network. First of all, the dragonfly algorithm is used to optimize the hyperparameters (epoch, learning rate and threshold) of the LSTM neural network. Then, the LSTM neural network prediction model is established through the obtained optimal parameters, and the Adam optimizer can be used to complete the training of other parameters of the network. Finally, the measured data collected by a wind farm in Texas is used to complete the short-term forecasting of wind power. Through comparative analysis, the DA-LSTM model proposed in this article has a better prediction effect than DA-BP, LSTM and ELM neural network.

2. Long Short-Term Memory
Long short-term memory neural network is a special type of recurrent neural network, which introduces the cell state and gating mechanism in the hidden layer of the original RNN. And they are used to save the long-term state and control the information transmission path in the neural network by LSTM. Therefore, LSTM can solve the problem of gradient disappearance and gradient explosion when RNN is dealing with long-term dependence problems to a certain extent [11].

LSTM neural network and RNN are similar in structure, both of them are consisted by input layer, hidden layer and output layer. The difference is that LSTM replaces the hidden unit of RNN with a memory cell with gating function by introducing input gate, forget gate and output gate [12], and Figure 1 shows its unit structure.

![Fig 1. The structure of LSTM neural network unit](image-url)
In the unit structure of the LSTM neural network, the forget gate determines whether the previous information is retained. This gate reads the output \( h_{t-1} \) of the memory cell at time \( t-1 \) and the input \( x_t \) at time \( t \). Under the action of the sigmoid activation function, it can output a value between 0 and 1. When the value is 0, it means that the information is completely discarded, and when the value is equal to 1, it is completely retained. The calculation formula is shown in Equation (1).

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

(1)

The input gate determines whether to write the current information to update the memory. Concretely, the sigmoid activation function in the input gate controls the content of the information that needs to be updated. The tanh activation function creates a new vector \( C'_t \) that will be added to the cell state by reading the value of \( h_{t-1} \) and \( x_t \). Its calculation formula is as shown in Equations (2)-(3).

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

(2)

\[
C'_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)
\]  

(3)

By multiplying the cell state \( C_{t-1} \) and \( f_t \) at \( t-1 \) to forget the unnecessary information, and then add the value of \( i_t \cdot C'_t \) to complete the update of the cell state, as shown in Equation (4).

\[
C_t = f_t \cdot C_{t-1} + i_t \cdot C'_t
\]  

(4)

After the cell state update is completed, the output gate needs to determine the content of the output information. It determines the output part of the cell state through the sigmoid function, uses the tanh function to process the updated cell state \( C_t \), and multiplies it with the output value \( o_t \) of the sigmoid function to obtain the output information \( h_t \) of the memory cell. The calculation process is shown in Equations (5)-(6).

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

(5)

\[
h_t = o_t \cdot \tanh(C_t)
\]  

(6)

3. The realization process based on the DA-LSTM prediction model

3.1. dragonfly optimization algorithm

The dragonfly algorithm is a new type of intelligent swarm optimization algorithm proposed by Seyedali Mirjalili in 2016 [13]. Its main inspiration is from the group hunting behavior of dragonflies in nature. The algorithm mathematically models the life habits of dragonflies, such as flight routes, avoiding natural enemies, and finding food. In nature, the hunting behavior of dragonflies mainly includes: separation, queuing, cohesion, hunting and avoiding natural enemies [14]. The mathematical model is as follows:

Separation is the behavior of a dragonfly individual separating from its neighbors, and its mathematical expression is shown in formula (7).

\[
S_i = -\sum_{j=1}^{N} (X - X_j)
\]  

(7)

Where \( S_i \) is the separation of the i-th dragonfly individual, \( X \) is the current location of the individual, \( X_j \) is the position of the j-th dragonfly adjacent to the individual, and \( N \) is the total number of individuals adjacent to the dragonfly \( X \).

Queuing is to keep one's own movement is consistent with the behavior of neighboring individuals, and its mathematical expression is shown in formula (8).

\[
A_i = \sum_{j=1}^{N} V_j / N
\]  

(8)
Where \( \mathbf{A}_i \) is the position vector of the \( i \)-th dragonfly individual when participating in the queuing behavior, and \( \mathbf{V}_j \) is the flight speed of the adjacent individual.

Cohesion is the behavior of dragonflies and their neighbors gathering together, and its mathematical expression is described in formula (9).

\[
C_i = \frac{\sum_{j=1}^{N} X_j - X}{N}
\]  
(9)

Where \( C_i \) is the position vector of the \( i \)-th dragonfly individual during the collective behavior.

Predation is the behavior of dragonflies looking for food in order to survive. Its mathematical expression is shown in formula (10).

\[
F_i = X^+ - X
\]  
(10)

Where \( F_i \) is the position vector of the \( i \)-th dragonfly individual during the hunting behavior, and \( X^+ \) is the position of the prey.

Escape from natural enemies, that is the behavior of dragonflies to prevent themselves from being preyed. Its mathematical expression is shown in formula (11).

\[
E_i = X^- + X
\]  
(11)

Where \( E_i \) is the position vector of the \( i \)-th dragonfly individual escaping from natural enemies, and \( X^- \) is the position of the individual hunting.

In the process of simulating the group behavior of dragonflies, the step length and position vector of its motion need to be updated, and the specific update method is shown in formula (12).

\[
\Delta X_{i+1} = (s \mathbf{S}_i + a \mathbf{A}_i + c \mathbf{C}_i + f \mathbf{F}_i + e \mathbf{E}_i) + \omega \Delta X_i
\]  
(12)

Where \( s \) is the separation weight, \( a \) is the queuing weight, \( c \) is the collection weight, \( f \) is the hunting weight, \( e \) is the escape weight, and \( \omega \) is the inertia weight.

In addition, most dragonflies in nature are in motion, so their position needs to be updated. The mathematical expression is shown in formula (13).

\[
X_{i+1} = X_i + \Delta X_{i+1}
\]  
(13)

When there is no neighboring individual around the dragonfly, \( \text{Lévy} \) flight needs to be used to update the position of the dragonfly. Its mathematical expression is shown in formula (14).

\[
X_{i+1} = X_i + \text{Lévy}(d) \times X_i
\]  
(14)

Where \( d \) represents the dimension of the individual position vector. The expression of the \( \text{Lévy} \) function is shown in formulas (15)-(16), and \( \beta \) in formula (16) is a constant.

\[
\text{Lévy}(x) = 0.01 \frac{\Gamma(\frac{d}{\beta})^* \sigma}{\Gamma(d)} \times r_1, r_2 = \text{rand}(0,1)
\]  
(15)

\[
\sigma = \sqrt{\frac{\Gamma(d + 1)}{\Gamma(d)}} \sqrt{\frac{d \sin(\frac{\pi \beta}{2})}{\Gamma\left(\frac{1 + \beta}{2}\right)}} \left(\frac{\beta \Gamma\left(\frac{d}{\beta}\right)}{2}\right)^{\frac{d}{\beta}} \times (x - 1)!
\]  
(16)

3.2. Data preprocessing

The volatility of the sample data will directly affect the accuracy of the prediction results and lead to the uncertainty of the prediction model, and the normalization of the data can reduce the impact of the dimension and value range differences between the features. Hence, the sample data needs to be normalized before its are input into the model for training. The normalization method used in this study is the discrete standardization method. This method utilizes the linear transformation of the
original data to map the data values to the range of [0,1], the specific calculating method is shown in
formula (17).

\[
y' = \frac{y - y_{\text{min}}}{y_{\text{max}} - y_{\text{min}}}
\]  

(17)

Where \( y' \) is the normalized wind power generation value, \( y_{\text{max}} \) and \( y_{\text{min}} \) are the maximum and minimum values of all wind power generation data, respectively, and \( y \) is the original wind power generation data.

After inputting the preprocessed data into the prediction model to complete the training of the model, the predicted value output by the prediction model needs to be denormalized, and the corresponding reduction function is shown in formula (18).

\[
y_{\text{pre}} = y_{\text{pre}} (y_{\text{max}} - y_{\text{min}}) + y_{\text{min}}
\]  

(18)

Where \( y_{\text{pre}} \) is the predicted value of wind power after restoration, and \( y_{\text{pre}} \) is the predicted value of wind power before restoration.

3.3. The DA-LSTM forecasting model

Wind power poses a challenge to the stable operation of the power system due to its intermittent and volatility, and short-term wind energy forecasting can effectively solve this problem. Therefore, the paper proposes a DA-LSTM-based forecasting model. The specific structure is shown in the figure 2.

---

Fig 2. The flowchart of DA-LSTM forecasting model
The specific modeling steps are depicted as follows:
Step 1: Divide the collected wind power data into training set and testing set at a ratio of 7:3, and then utilize the training set as the input of the prediction model to train the model.
Step 2: Use the discrete standardization method to normalize all sample data.
Step 3: Initialize the parameters of the prediction model. Set the search range of the epoch, threshold, the learning rate. Set the value of the maximum number of iterations and the population of the dragonfly algorithm.
Step 4: According to the initial epoch, learning rate and threshold value, the LSTM model is constructed, then the training set data will be used to train the model, and the root mean square error of the output value will be used as the fitness value of the individual.
Step 5: Update the position of food and natural enemies according to Eqs. (7)-(11), and then use Eqs. (12)-(14) to update the individual’s movement step length and position vector, calculate the corresponding dragonfly individual fitness value, and compare the local and the global optimal solution to make the prediction effect best.
Step 6: Judge whether the individual meets the termination condition, if so, output the optimal epoch, learning rate and threshold, otherwise return to step (4).
Step 7: Utilize the optimal hyperparameters to construct a DA-LSTM prediction model. After inputting the testing set data to the model, use the real value and the prediction data output by the model to evaluate the prediction effect of the model combined with the model’s evaluation index.

3.4. The model performance evaluation index
In order to accurately evaluate the performance of the DA-LSTM prediction model, the study selects the root mean square error (RMSE), mean absolute error (MAE) and coefficient of determination as evaluation indicators $R^2$ to analyze the prediction accuracy of the model. Among them, the smaller the value of RMSE and MAE, the closer the value of $R^2$ to 1, the higher the prediction accuracy of the model, the better the prediction effect. The specific definition formulas as Eqs. (19)-(21) shown.

\[
RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_{\text{true}}(i) - y_{\text{pre}}(i))^2} \tag{19}
\]
\[
MAE = \frac{1}{n} \sum_{i=1}^{n} |y_{\text{true}}(i) - y_{\text{pre}}(i)| \tag{20}
\]
\[
R^2 = 1 - \frac{\sum_{i=1}^{n} (y_{\text{true}}(i) - y_{\text{pre}}(i))^2}{\sum_{i=1}^{n} (y_{\text{true}}(i) - \bar{y}_{\text{true}})^2} \tag{21}
\]

Where $n$ is the sample size of the testing set, $y_{\text{true}}(i)$ and $y_{\text{pre}}(i)$ are the true and predicted values of wind power, respectively, and $\bar{y}_{\text{true}}$ is the average value of the real wind power data.

4. Analysis of examples
4.1. Experimental environment and experimental data description
The construction and training of the DA-LSTM prediction model proposed in this article is based on the TensorFlow deep learning framework. The experimental software platform is Anaconda. And the hardware platform is the workstation configured with two CPUs (Intel Xeon E5-1600 and Intel Xeon E5-2600) and two high-end graphics cards (both of them are NVIDIA RTX 2080Ti). Besides, the hardware platform is equipped with Windows 10 Pro systems and 64G RAM.

A total of 8928 time series data used in this study is the measured data of a wind farm in Texas provided by the National Renewable Energy Laboratory (NREL) in January 2012. And the sampling interval of the data is 5 minutes.
4.2. Parameter selection and model construction
The parameter selection of the forecasting model is very important to the reliability of the model. The paper uses the preprocessed training set data, takes the root mean square error as the fitness function, uses the dragonfly algorithm to optimize, and finally obtains the optimal value of the three key hyperparameters (learning rate, iteration period, and threshold) in the LSTM neural network. The setting of each parameter in the optimization process is shown in Table 1.

The DA-LSTM neural network prediction model used in this study consists of an input layer, a double-layer hidden layer and an output layer. The hidden layer takes sigmoid as the activation function of the neuron, and the number of neurons in the hidden layer is set to 200. Other internal parameters are trained by Adam algorithm. At the same time, in order to test and illustrate the validity and accuracy of the DA-LSTM prediction model, the paper uses three neural networks (GWO-BP, ELM, and LSTM) to compare with the model. Four different models will be used to predict wind power separately and compare the predictions effect.

Table 1. DA-LSTM model parameters

| Parameters                  | Value           |
|-----------------------------|-----------------|
| Numbers of population      | 50              |
| Max iteration               | 100             |
| The range of learning rate  | [0.002,0.001]   |
| The range of threshold      | [0.8,1.5]       |
| The range of epoch          | [50,300]        |

4.3. Analysis of results
In this article, the DA-LSTM prediction model is used to predict short-term wind power. The actual value curve of wind farm output power and the prediction curve of each model are shown in Figure 3. Figure 4 shows the absolute value error of each model prediction. It can be seen from Figure 5 that the absolute value error of the prediction of the DA-LSTM model is smaller than other models, and the prediction effect of the model is better. In order to better evaluate the prediction performance of different models, the paper introduces three evaluation indicators, namely the root mean square error (RMSE), the average absolute error (MAE), and the coefficient of determination ($R^2$) to evaluate the prediction accuracy of each model.

Using the testing set data, the prediction errors of different models can be obtained. The calculation results are shown in Table 2. According to the comparative analysis of the data results in Table 2, compared with the GWO-BP and ELM models, the RMSE of the LSTM model is reduced by 24.58% and 39.69%, and the MAE of LSTM is reduced by 29.69% and 43.39% respectively. According to these, LSTM neural network is more effective in dealing with time series-related forecasting problems. Compared with the LSTM model, the RMSE of the DA-LSTM model is reduced by 38.49%, and the MAE is reduced by 34.07%, which shows that the DA-LSTM model constructed after optimizing the hyperparameters of the LSTM is better than the original model, and the prediction accuracy is higher. Moreover, the $R^2$ of the DA-LSTM model reaches 0.9977, which is closest to 1 among the value of four different prediction models, indicating that the model has the highest degree of fitting the testing set data and the prediction effect of the model is the best. Experimental results show that compared to ELM, GWO-BP and LSTM models, DA-LSTM model is more suitable for short-term wind power prediction, and it has higher prediction accuracy and reliability.

Table 2. Comparison of prediction error results of various models

| Model  | RMSE  | MAE  | $R^2$ |
|--------|-------|------|-------|
| ELM    | 0.5769| 0.4287| 0.9883|
| GWO-BP | 0.4613| 0.3452| 0.9742|
5. Conclusion
In order to improve the accuracy of short-term wind power forecasting, the paper proposes a DA-LSTM forecasting model to predict wind power generation based on the good performance of long...
and short-term memory neural networks in time series forecasting, and obtains the following conclusions:

1) The dragonfly algorithm is used to optimize the hyperparameters of the LSTM neural network, which effectively solves the problem of low prediction accuracy of the model caused by selecting parameters based on experience.

2) This paper proposes the DA-LSTM prediction model based on the excellent performance of using LSTM neural network to process long-term series data. Compared with the other three models, the RMSE and MAE of the model are lower, and the $R^2$ is higher, which indicate the model has higher prediction accuracy and reliability.

3) With the rapid development of computer technology, optimization algorithm and deep learning, other optimization algorithms can be considered to optimize the hyperparameters of the LSTM neural network and more novel neural networks should be introduced, which can further improve the accuracy of the prediction model. In addition, we can also explore the application of the prediction model proposed in this article to other fields. In this case, we are able to dig deeper into the prediction potential of the model.

References

[1] Mujeeb, S., Alghamdi, T. A., Ullah, S., Fatima, A., Javaid, N., Saba, T. (2019) Exploiting deep learning for wind power forecasting based on big data analytics. J. Applied Sciences, 9: 4417.

[2] Zhu, C. X., Zhang, Y., Yan, Z., Zhu, J. D., Zhao, T. (2020) A wind power time series modeling method based on the improved Markov Chain Monte Carlo Method. J. Transactions of China Electrotechnical Society, 35: 577-589.

[3] Xiang, L., Deng, Z., Hu, A. (2019) Forecasting short-term wind speed based on IEWT-LSSVM model optimized by bird swarm algorithm. J. IEEE Access, 7: 59333-59345.

[4] López, E., Valle, C., Allende, H., Gil, E., Madsen, H. (2018) Wind power forecasting based on echo state networks and long short-term memory. J. Energies, 11: 526.

[5] Yu, R., Gao, J., Yu M., Lu, W., Xu, T., Zhao, M., Zhang, J., Zhang, R. X., Zhang, Z. (2019) LSTM-EFG for wind power forecasting based on sequential correlation features. J. Future Generation Computer Systems, 93: 33-42.

[6] Du, P., Wang, J., Yang, W., Niu, T. (2019) A novel hybrid model for short-term wind power forecasting. J. Applied Soft Computing, 80: 93-106.

[7] Cao, J. B., Ren, Z. J., Deng, X. H., Fan, W. S., Liu, L. J., Sun, J. G. (2019) Wind power forecast considering differential times of optimal ARIMA model. J. Proceedings of the CSU-EPSA, 31: 105-111.

[8] Liu, A. G., Xue, Y. T., Hu, J. L., Liu, L. P. (2015) Ultra-short-term wind power forecasting based on SVM optimized by GA. J. Power System Protection and Control, 43: 90-95.

[9] Ma, L. X., Wu, L. (2018) Prediction model of wind power based on lifting wavelet transform. J. Electric Power Science and Engineering, 34: 20-25.

[10] Li, X. J., Xu, G. J. (2019) Wind power prediction method based on long short-term memory neural network. J. Power Generation Technology, 40: 426-433.

[11] Lang, W., Ma, X., Luo, Y., Zhou, B., Yang, D. (2019) Wind power prediction based on principal component analysis and long short-term memory networks. In: IEEE Innovative Smart Grid Technologies-Asia (ISGT Asia). Chengdu. pp. 3063-3068.

[12] Yu, Y., Si, X., Hu, C., Zhang, J. (2019) A review of recurrent neural networks: LSTM cells and network architectures. J. Neural computation, 31: 1235-1270.

[13] Mirjalili, S. (2016) Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. J. Neural Computing and Applications, 27: 1053-1073.
[14] Amroune, M., Bouktir, T., Musirin, I. (2018) Power system voltage stability assessment using a hybrid approach combining dragonfly optimization algorithm and support vector regression. J. Arabian Journal for Science and Engineering, 43:3023-3036.