



Short-Term Forecasting of Wind Power Considering Parameter Optimization

Songyuan Liu¹, Yu Meng², Rui Guo², Qiankun Wang¹, Jinming Yu¹,
and Feilong Fan¹ (✉)

¹ Shanghai Jiao Tong University, Shanghai 200240, China
feilongfan@sjtu.edu.cn

² Shanghai Electric Power Design Institute Co., Ltd., Shanghai 200025, China

Abstract. Wind power is a major form of renewable energy and has significant development potential. To deal with the wind power volatility, we need accurate wind power forecasting methods. This article presents a comprehensive forecasting method combining Convolution Neural Network (CNN) and Long Short-Term Memory (LSTM) Recurrent Neural Network, and uses differential evolution algorithm (DE) for parameter optimization. First, CNN is used to extract high-dimensional features, and then the LSTM model is used for time series prediction. The differential evolution algorithm is used to adjust the parameters to achieve better prediction results. The testing based on the data from a practical wind plant shows that the prediction results using the proposed forecasting method is more accurate than the prediction results using the existing forecasting method.

Keywords: Wind power · Comprehensive forecasting · Convolution neural network · Long short-term memory · Differential evolution algorithm

1 Introduction

With worldwide energy shortage and environmental problems, the use of renewable energy is becoming more of a concern [1, 2]. As a clean, pollution-free, widely distributed renewable energy that is easy to exploit, wind energy has received widespread attention [3, 4].

However, wind energy is easily influenced by wind speed and air pressure, leading to strong fluctuation and intermittence of wind power, which harms its integration into the power grid and hinders its development [5, 6]. Accurate and reliable short-term wind power forecasting can not only help wind farms in work planning, but also reduce the impact on the power grid, which is an effective way to overcome this obstacle.

With the progress of research, wind speed forecasting methods are constantly being optimized and updated, from the physical method at the beginning to the statistical method, and in recent years, intelligent algorithms are mainly used for data analysis [7]. Physical methods are usually based on complex models, and make predictions according to all kinds of meteorological data. This method is suitable for medium and long-term wind forecasting, but performs not so well in short-term wind power forecasting [8].

Traditional statistical methods use historical data of wind power to forecast, the most popular model is auto-regressive integrated moving average (ARIMA) [9]. It predicts well on the linear part of the wind power fluctuation, but it cannot learn nonlinear characteristics of the change of wind power, so it fails to achieve a long-term satisfactory effect.

A study on recurrent neural network (RNN) finds it have higher accuracy compared with ARIMA method [10]. Many other neural networks such as Deep belief network (DBF) [11], convolution neural network (CNN) [13], long short-term memory (LSTM) [14] and echo state network (ESN) [12] are also valued for their advantages in feature extraction and nonlinear fitting. Among these models, the LSTM model is promising due to its advantages in processing long time series [15]. Hybrid model are also proposed for wind speed forecasting [16]. For example, for the LSTM network, empirical wavelet transforms [17], ensemble empirical mode decomposition [18] are respectively combined into the model for pre-optimization of data. For the parameters in the neural network model, many methods such as differential evolution algorithm (DE) [19], grey wolf optimization (GWO) [20], multi-objective whale optimization algorithm (MOWOA) [21] have been proposed. This is because network parameters, if not properly selected, will have a bad influence on the prediction results [22]. Another problem of those networks is when the input wind power fluctuates greatly, the prediction accuracy will be greatly reduced.

To find features under long-term input sequence and smooth the data, we first use CNN to extract local informative features in the input in our network. LSTM is often used for time series data prediction due to its excellent performance in long-term and relevant data processing [23]. Therefore, we next use LSTM to perform time series prediction on non-linear wind power data. Considering that there is currently no clear algorithm to guide the selection of parameters in the neural network, we use a reliable improved differential evolution algorithm to optimize parameters such as the number of layers and forgetting rate of the LSTM network to achieve better prediction results.

2 Strategy for Wind Power Forecasting

In order to achieve smart operation of wind farms and predict the wind power in advance to better arrange the start and stop of wind turbines in wind farms, neural network methods are used for short-term wind power forecasting. They can achieve better forecast results than traditional methods based on the historical data of wind power in a period of time, providing a basis for the subsequent optimal control.

Wind power prediction mostly adopts the LSTM model. It is suitable for time series prediction, but sensitive to fluctuations, and if the network parameters are not properly selected, the prediction accuracy is often low. Therefore, a comprehensive method of integrating CNN, LSTM and DE is proposed here. First, we use CNN to smooth the data and better extract high—dimensional features, then use LSTM network to predict the result. Differential evolution algorithm is used to optimize the parameters of the network (Fig. 1).

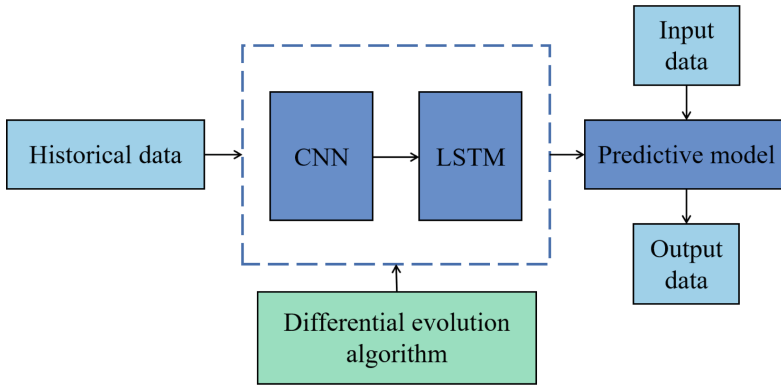


Fig. 1. Structure of CNN-LSTM-DE method

3 CNN-LSTM Structure

3.1 CNN

CNN is a successful deep learning architecture and it is an effective for feature extraction and pattern recognition [24]. It can extract high-dimensional features of input data through convolution layer (Fig. 2).

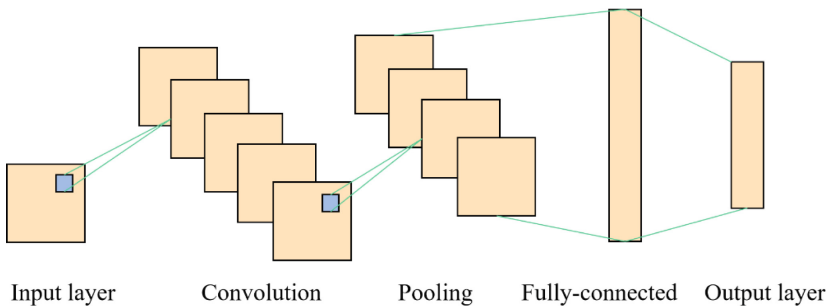


Fig. 2. Structure of CNN

The data input from the input layer is convolved, pooled, fully connected, and then output through the output layer in CNN network [27]. The function of convolution layer is to locally perceive input data, and extract the high-dimensional features of the input through the product of convolution kernel and sliding window. After the convolution layer is the pooling layer, also called the convergence layer. Convolution layer reduces the number of connections between different neurons, but the number of neurons does not decrease significantly. So the pooling layer is required to participate, which is used to reduce feature dimensions and avoid over fitting [28]. The last step is the fully connected layer to output the results. After the data is processed by the convolution layer and pooling, it is input to the fully connected layer to get the final result. After such a series

of operations, the data volume has decreased significantly, the efficiency has improved significantly, and the computing cost has also decreased significantly. The most important convolution operation in CNN is shown in formula (1):

$$O_t = \sigma(W * x_t + b) \quad (1)$$

where W represents the weight coefficient of the filter; x_t represents the data information of the input sample at time t ; $*$ represents the discrete convolution operation between x_t and W ; b is the bias parameter, which is obtained by learning and will be passed when training the model; σ represents the activation function; O_t represents the output data after the convolution operation.

Each neuron contains a filter. The input data will be convolved with each filter, and the results will be added together as the output data. For example, when a sequence is all 2, the filter (1, 1) is used, the step size of the sliding window is 1, then the output is 4 if there is no bias parameter.

3.2 LSTM

LSTM network is a variant of Recurrent Neural Network (RNN). RNN is very effective for data with sequence characteristics. It can mine temporal information in data [25], but in the process of model training, gradient disappearance or gradient explosion will occur along with the accumulation of time steps. This will cause the loss of previous historical input information or generate invalid information, resulting in wrong prediction results.

The LSTM network selectively store information by introducing forget gate, input gate and output gate to one cell to control the transmission of information, solving the

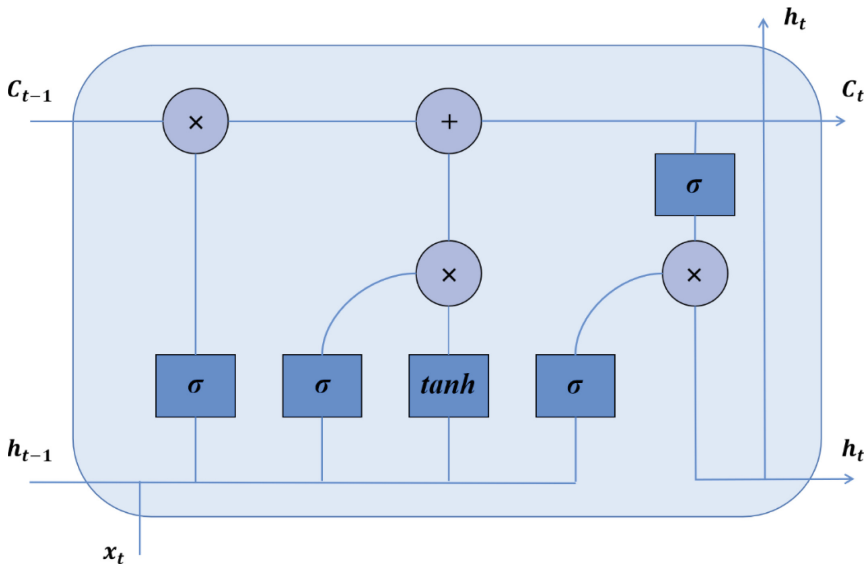


Fig. 3. LSTM cell structure

problem of gradient disappearance or gradient explosion in long sequences in RNN to a certain extent [26]. The cell structure of LSTM is shown in Fig. 3.

The left end of the LSTM network structure in the figure is the input layer, the right end is the output layer, and the middle part is three gating units. First go through the input gate to see whether there is information input, then judge whether the forgetting gate chooses to forget the information in the memory cell, and finally go through the output gate to judge whether to output the information at this moment.

The forget gate is used for deletion of memory. First, combine h_{t-1} with x_t , and then use the W_f matrix to adjust it to the same dimension as the hidden layer at time t . The next step is to put them into the sigmoid function. Finally, use the sigmoid function to compress the output value between 0 and 1. Output values close to 0 will be eliminated, and values close to 1 will be retained. The operation formula is shown in formula (2).

$$f_t = \sigma(W_f \cdot h_{t-1} + W_f \cdot x_t + b_f) \quad (2)$$

In the formula, f_t represents the past memory measurement factor, W_f represents the weight, b_f represents the bias, h_{t-1} represents the state information of the previous hidden layer, and x_t represents the input vector at time t .

$$i_t = \sigma(W_i \cdot h_{t-1} + W_i \cdot x_t + b_i) \quad (3)$$

$$k_t = \tan h(W_k \cdot h_{t-1} + W_k \cdot x_t + b_k) \quad (4)$$

$$c_t = f_t * c_{t-1} + i_t * k_t \quad (5)$$

The input gate is used to update the information memory. First, h_{t-1} and x_t are put into the sigmoid function for information screening. Meanwhile, h_{t-1} and x_t are passed to the tanh function in order to create a new candidate value vector, then calculate the output value of the sigmoid function with the output value k_t of the tanh function and add it to the past memory information c_{t-1} to expand the memory capacity, Finally, add the product of f_t and c_{t-1} to get the updated memory c_t .

$$g_t = \sigma(W_g \cdot h_{t-1} + W_g \cdot x_t + b_g) \quad (6)$$

$$h_t = g_t * \tan h(c_t) \quad (7)$$

The output value of the output gate shall be determined according to the cell state. First, we put h_{t-1} and x_t into a sigmoid function to determine which part of the cell state needs to be output, then process the cell state through the tanh layer, and multiply the two to get the final information we want to output.

4 Differential Evolution Algorithm

Differential evolution algorithm (DE) is a heuristic search algorithm, which adopts the evolution law of “survival of the fittest” to conduct random search. Under the global search strategy of preserving samples, mutation operation, crossover operation and one-to-one competition, survival principle are adopted according to the difference of vector between parent samples, which improves local search ability, robustness and convergence.

Using the method of random selection, in the n -dimensional space, M sample data are generated to satisfy the following constraints. In the search space, it is assumed that x_{ij}^U is the upper bound of the space search of the i th sample in the j th dimension space, while x_{ij}^L is the lower bound. Samples that meet the constraints within its range are initialized.

$$x_{ij}(0) = \text{rand}_{ij}(0, 1) \left(x_{ij}^U - x_{ij}^L \right) + x_{ij}^L \quad (8)$$

In the formula, $x_{ij}(0)$ represents the initialization sample.

When performing the mutation operation, the mutation factor F is introduced to control the ethnic diversity and convergence. In the traditional differential evolution algorithm, the value range of F is $[0, 2]$. When the value of F is small, it will not necessarily break through its local extremum during the evolution process, resulting in premature convergence. If the value of F is large, it will easily jump out of the local extreme value; however, the speed of convergence is reduced as well.

The mutation vector and a predetermined target vector are mixed with parameters to generate a test vector in the mutation operation, and the global and local area searches are balanced by controlling the numerical value of individual parameters in each dimension. Compare the test vector with the original vector, and choose the better one as the new solution vector, update the vector, and proceed to the next step. The traditional structure of DE is shown in the figure below (Fig. 4).

This paper improves the traditional differential evolution algorithm in the following two aspects. The first improvement is that two pairs of popular mutation operators are used in the mutation stage of the search process. In each generation, the selection depends on the absolute error between the best value and the mean value of the objective function according to the previous generation. When the error is large in earlier generations, one pair of mutation operator are used to explore the scope containing the global optimal solution. When error is smaller after several rounds of global search, another pair of mutation operator will be used to increase the algorithm convergence speed [29].

Another improvement can be called elite selection technology. The test vectors generated after crossover operation are mixed with their parent population. Then select the best individual from the whole group to build a new group for the next generation. In this way, the best individuals in the whole population are passed on to the next generation, which enables the differential evolution algorithm achieve better convergence speed.

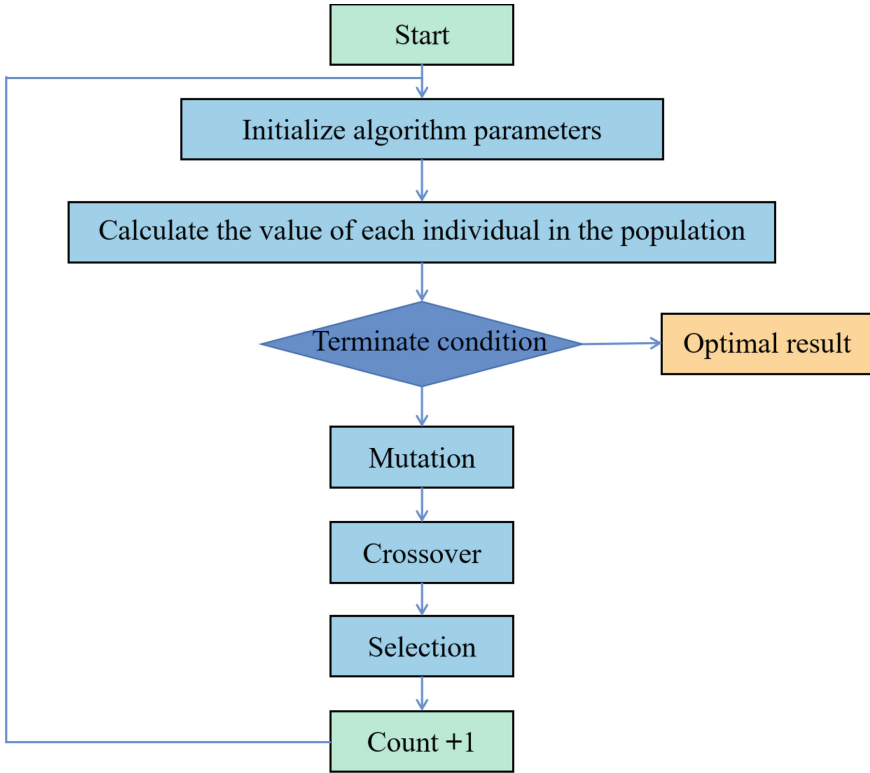


Fig. 4. Flow chart of DE algorithm

5 Case Study

Applying the CNN-LSTM-DE method to perform short-term wind power prediction on wind farms. The experimental data comes from the wind power data of an offshore wind farm from January 1st 2020 to January 16th 2020, with sampling intervals of 10 min. The data only has one column of wind power over time. The original wind power output of the farm is shown in the figure below.

Symmetric mean absolute percentage error (SMAPE) is used to evaluate the prediction. It is a correction based on MAPE, which can better avoid the problem that the calculation result of MAPE is too large when the real value is small. The formula is as follow.

$$SMAPE = \frac{1}{n} \sum_{t=1}^n \frac{|F_t - A_t|}{(A_t + F_t)/2} \tag{9}$$

where A_t is the actual value and F_t is the forecast value. When the predicted F_t and the real A_t are exactly the same, the minimum SMAPE value is 0, so the prediction is better when its value of SMAPE closer to 0.

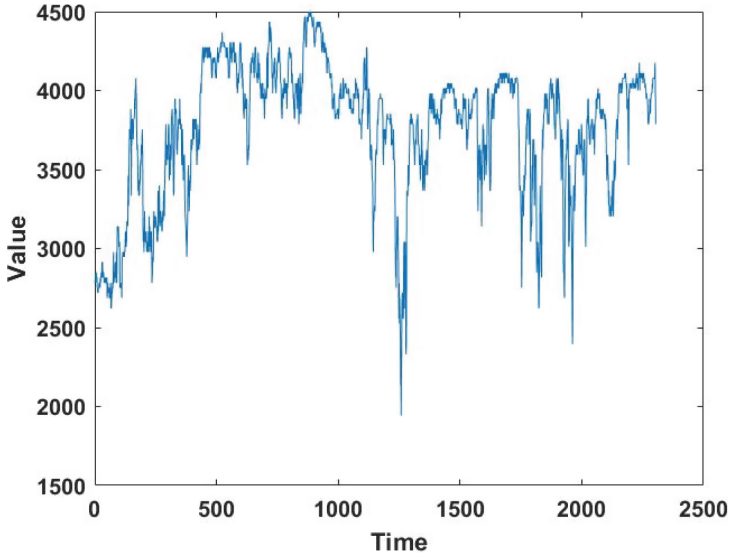


Fig. 5. Original wind farm output power sequence

First, we study the case where the differential evolution algorithm is not used for optimization. We arrange the hyperparameters in the LSTM network by convention. The LSTM network has two layers, the number of neurons in the first and second layer is 64 and 16 respectively. The batch size is 12, learning rate is 0.01 and dropout proportion for 2 layers are both 0.5. The difference between the predicted value and the actual value is shown in the following figure and we can calculate the SMAPE of it is 0.0165.

Then, we applied the differential evolution algorithm to the original model and optimized the parameters of the LSTM. The number of neurons in the two layers is 176 and 124 respectively while the dropout proportion is 0.442 and 0.809. The batch size is 14 and learning rate decreases to 0.000696. The difference between the predicted value and the actual value is shown in Fig. 5 and 6 and we can calculate the SMAPE of it is 0.0109 (Figs. 7 and 8).

Comparing the CNN-LSTM-DE model with the CNN-LSTM model that is not optimized by the differential evolution algorithm, it can be found that the prediction results come from the optimized model will be closer to the actual value. This shows the great effect of optimization algorithms on model improvement.

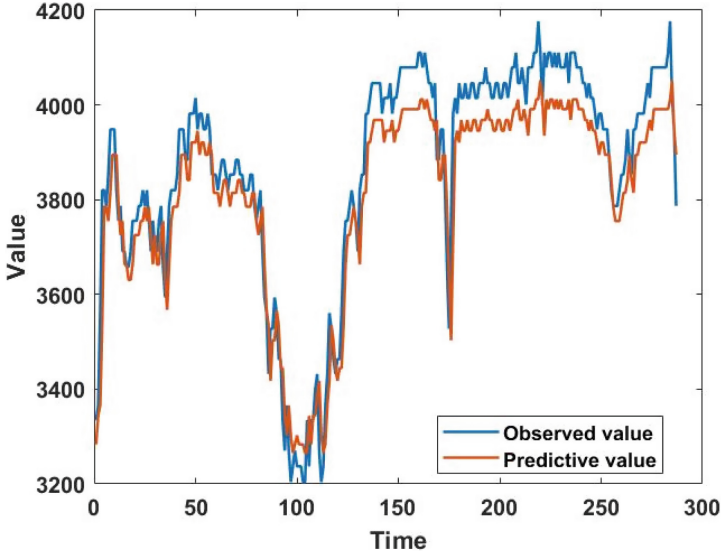


Fig. 6. Comparison of CNN-LSTM model predicted value and actual value

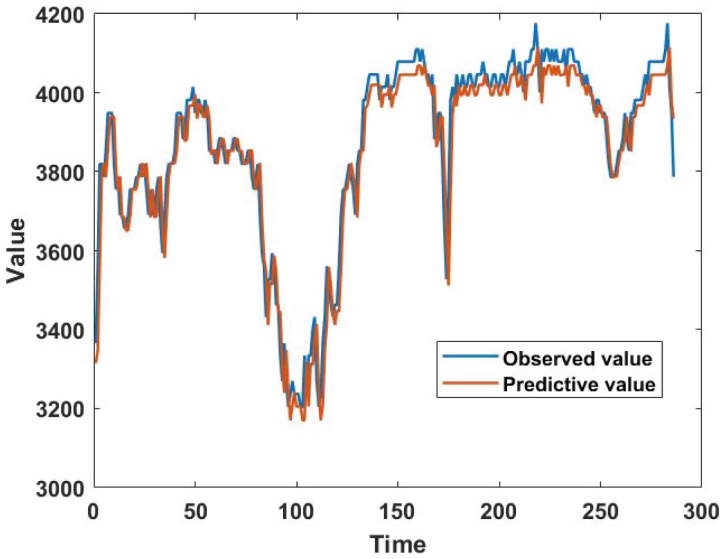


Fig. 7. Comparison of CNN-LSTM-DE model predicted value and actual value

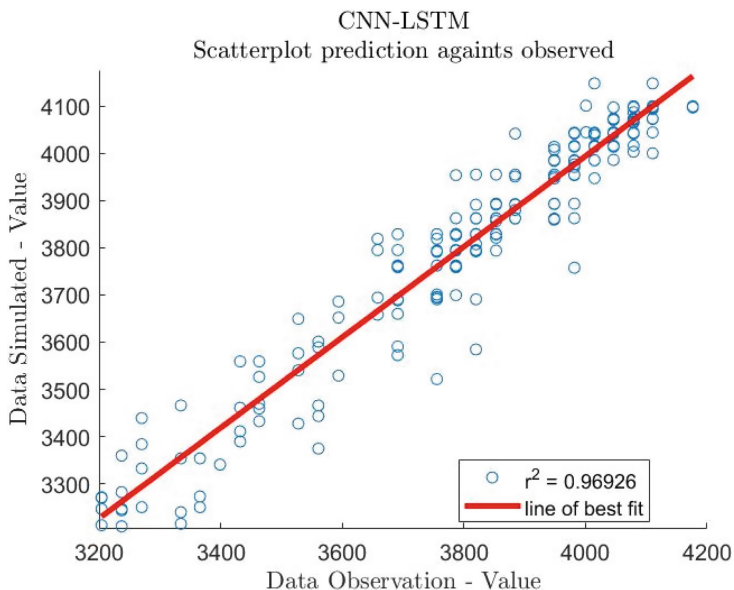


Fig. 8. CNN-LSTM-DE scatterplot prediction against observed

6 Conclusion

This paper proposes a combined CNN-LSTM-DE method for wind power short-term data forecasting. For the existing past data, first use CNN for feature extraction, and then use LSTM network for prediction, making full use of the historical data. In order to improve the accuracy of the model, an improved differential evolution algorithm is used to optimize parameters such as number of layers in the network, number of neurons in each layer and the learning rate of the network. Tests on the offshore wind power dataset show that the network model using the optimization algorithm has higher prediction accuracy. The reliable performance of the model suggests that we can use this structure in a variety of time series forecasting, especially with regard to power forecasting.

Acknowledgment. This research is supported by Research on Key Technologies of Wind-solar Digital Twin Integration KJ-2021-106, National Key Research and Development Program 2019YFE0102900 and Innovation Program of Shanghai Municipal Education Commission 2019-01-07-00-02-E00044.

References

1. Kılıç, Ş, Krajačić, G., Duić, N., et al.: Research frontiers in sustainable development of energy, water and environment systems in a time of climate crisis. *Energy Convers. Manage.* **199**, 111938 (2019)
2. Fan, F., Huang, W., Tai, N., et al.: A multilevel overvoltage prevention strategy for the distribution networks with high penetration of rooftop photovoltaic systems. *Int. Trans. Electric Energy Syst.* **29**(7), e12046 (2019)

3. Vargas, S.A., Esteves, G.R.T., Maçaira, P.M., et al.: Wind power generation: a review and a research agenda. *J. Clean. Prod.* **218**, 850–870 (2019)
4. Fan, F., Zhang, R., Xu, Y., et al.: Robustly coordinated operation of an emission-free microgrid with hybrid hydrogen-battery energy storage. *CSEE J. Power Energy Syst.* **8**(2), 369–379 (2021)
5. Nikolić, V., Sajjadi, S., Petković, D., et al.: Design and state of art of innovative wind turbine systems. *Renew. Sustain. Energy Rev.* **61**, 258–265 (2016)
6. Fan, F., Tai, N., Zheng, X., et al.: Equalization strategy for multi-battery energy storage systems using maximum consistency tracking algorithm of the conditional depreciation. *IEEE Trans. Energy Convers.* **33**(3), 1242–1254 (2018)
7. Liu, H., Chen, C.: Data processing strategies in wind energy forecasting models and applications: a comprehensive review. *Appl. Energy* **249**, 392–408 (2019)
8. Jung, J., Broadwater, R.P.: Current status and future advances for wind speed and power forecasting. *Renew. Sustain. Energy Rev.* **31**, 762–777 (2014)
9. Che, J., Wang, J.: Short-term electricity prices forecasting based on support vector regression and auto-regressive integrated moving average modeling. *Energy Convers. Manage.* **51**(10), 1911–1917 (2010)
10. Liu, H., Mi, X., Li, Y.: Wind speed forecasting method based on deep learning strategy using empirical wavelet transform, long short-term memory neural network and Elman neural network. *Energy Convers. Manage.* **156**, 498–514 (2018)
11. Wang, K., Qi, X., Liu, H., et al.: Deep belief network-based k-means cluster approach for short-term wind power forecasting. *Energy* **165**, 840–852 (2018)
12. Chitsazan, M.A., Fadali, M.S., Trzynadlowski, A.M.: Wind speed and wind direction forecasting using echo state network with nonlinear functions. *Renew. Energy* **131**, 879–889 (2019)
13. Pan, X., Wang, L., Wang, Z., et al.: Short-term wind speed forecasting based on spatial-temporal graph transformer networks. *Energy* **253**, 124095 (2022)
14. Han, L., Jing, H., Zhang, R., et al.: Wind power forecast based on improved Long Short-Term Memory network. *Energy* **189**, 116300 (2019)
15. Demolli, H., Dokuz, A.S., Ecemis, A., et al.: Wind power forecasting based on daily wind speed data using machine learning algorithms. *Energy Convers. Manage.* **198**, 111823 (2019)
16. Hu, Y.L., Chen, L.: A nonlinear hybrid wind speed forecasting model using LSTM network, hysteretic ELM and Differential Evolution algorithm. *Energy Convers. Manage.* **173**, 123–142 (2018)
17. Jiang, Y., Liu, S., Zhao, N., et al.: Short-term wind speed prediction using time varying filter-based empirical mode decomposition and group method of data handling-based hybrid model. *Energy Convers. Manage.* **220**, 113076 (2020)
18. Chen, Y., Dong, Z., Wang, Y., et al.: Short-term wind speed predicting framework based on EEMD-GA-LSTM method under large scaled wind history. *Energy Convers. Manage.* **227**, 113559 (2021)
19. Yang, W., Wang, J., Lu, H., et al.: Hybrid wind energy forecasting and analysis system based on divide and conquer scheme: A case study in China. *J. Clean. Prod.* **222**, 942–959 (2019)
20. Altan, A., Karasu, S., Zio, E.: A new hybrid model for wind speed forecasting combining long short-term memory neural network, decomposition methods and grey wolf optimizer. *Appl. Soft Comput.* **100**, 106996 (2021)
21. Wang, J., Du, P., Niu, T., et al.: A novel hybrid system based on a new proposed algorithm—multi-Objective Whale Optimization Algorithm for wind speed forecasting. *Appl. Energy* **208**, 344–360 (2017)
22. Chen, J., Zeng, G.Q., Zhou, W., et al.: Wind speed forecasting using nonlinear-learning ensemble of deep learning time series prediction and extremal optimization. *Energy Convers. Manage.* **165**, 681–695 (2018)

23. Neshat, M., Nezhad, M.M., Abbasnejad, E., et al.: A deep learning-based evolutionary model for short-term wind speed forecasting: A case study of the Lillgrund offshore wind farm. *Energy Convers. Manage.* **236**, 114002 (2021)
24. LeCun, Y., Bottou, L., Bengio, Y., et al.: Gradient-based learning applied to document recognition. *Proc. IEEE* **86**(11), 2278–2324 (1998)
25. Senjyu, T., Yona, A., Urasaki, N., et al.: Application of recurrent neural network to long-term-ahead generating power forecasting for wind power generator. In: 2006 IEEE PES Power Systems Conference and Exposition, pp. 1260–1265. IEEE (2006)
26. Yu, Y., Si, X., Hu, C., et al.: A review of recurrent neural networks: LSTM cells and network architectures. *Neural Comput.* **31**(7), 1235–1270 (2019)
27. Hu, A., Wu, L., Huang, J., et al.: Recognition of weld defects from X-ray images based on improved convolutional neural network. *Multimedia Tools Appl.* **81**(11), 15085–15102 (2022)
28. Zhang, T., Liu, Y., Liu, X., et al.: Research on gas turbine early warning based on deep learning. In: 2022 4th International Conference on Communications, Information System and Computer Engineering (CISCE), pp. 505–508. IEEE (2022)
29. Ho-Huu, V., Vo-Duy, T., Luu-Van, T., et al.: Optimal design of truss structures with frequency constraints using improved differential evolution algorithm based on an adaptive mutation scheme. *Autom. Constr.* **68**, 81–94 (2016)