



# Short-term wind speed forecasting using recurrent neural networks with error correction

Jikai Duan <sup>a</sup>, Hongchao Zuo <sup>a,\*</sup>, Yulong Bai <sup>b</sup>, Jizheng Duan <sup>c</sup>, Mingheng Chang <sup>a</sup>, Bolong Chen <sup>a</sup>

<sup>a</sup> College of Atmospheric Sciences, Lanzhou University, Lanzhou, 730000, China

<sup>b</sup> College of Physics and Electronic Engineering, Northwest Normal University Lanzhou, 730070, China

<sup>c</sup> Institute of Modern Physics, Chinese Academy of Sciences, Lanzhou, 730000, China

## ARTICLE INFO

### Article history:

Received 17 September 2020

Received in revised form

4 November 2020

Accepted 18 November 2020

Available online 24 November 2020

### Keywords:

Wind speed forecasting

Recurrent neural network

ARIMA

## ABSTRACT

As a type of clean energy, wind energy has been effectively used in power systems. However, due to the influence of the atmospheric boundary layer, wind speed exhibits strong nonlinearity and non-stationarity. Therefore, the accurate and stable prediction of wind speed is highly important for the security of the power grid. To improve the forecasting accuracy, a novel hybrid forecasting system is proposed in this paper that includes effective data decomposition techniques, recurrent neural network prediction algorithms and error decomposition correction methods. In this system, a novel decomposition approach is used to first decompose the original wind speed series into a set of subseries, then it predicts the wind speed by recurrent neural network, and finally, it decomposes the error to correct the previously predicted wind speed. The effectiveness of the proposed model is verified using data from four different wind farms in China. The results show that the proposed hybrid system is superior to other single models and traditional models and realizes highly accurate prediction of wind speed. The proposed system may be a useful tool for smart grid operation and management.

© 2020 Elsevier Ltd. All rights reserved.

## 1. Introduction

With the continuous development of society, the demand for energy has also increased with the increasing economic development. This has led to an increasingly serious crisis of the exhaustion of the limited and nonrenewable fossil energy reserves. At the same time, the extensive use of fossil energy in the past hundred years has led to global warming, haze and other environmental problems that have become increasingly prominent [1]. To meet these challenges, there has been a global effort to actively develop clean, efficient and sustainable renewable energy source to support future economic development. Currently, the widely used new energy sources are solar energy, wind energy, tidal energy, nuclear energy, and geothermal energy. Among these, wind energy has been widely used worldwide because of its wide distribution, high energy content, lack of toxicity and sustainability [2]. Compared with the traditional form of power generation, wind

power generation has the advantages of zero pollution and low operating cost and has become one of the fastest growing renewable energy power generation technologies in the world. According to a report released by the Global Wind Energy Association (GWEC) in 2019, the newly installed wind power capacity in the world in 2019 was 60.4 GW, an increase of 19% from 2018 that was the second highest wind power installed capacity in history. The total global wind energy capacity has now exceeded 651 million KW, an increase of 10% over 2018. 2020 is expected to be a record year for wind energy, and GWEC predicts that there will be 76 million kilowatts of new capacity [3]. It is believed that the wind power industry still has much room for future development.

Although wind energy has obvious advantages over other energy sources, due to the influence of different underlying surfaces in the atmospheric boundary layer, wind has the characteristics of chaos, randomness and intermittence, making the temporal and spatial distribution of wind speed data more complex. This gives rise to great difficulties in the prediction of wind speed [4,5]. However, this challenge has not stopped human beings from pursuing the maximum utilization of wind energy. In recent decades, to improve the accuracy of wind speed forecasting, a large number of

\* Corresponding author.

E-mail address: [zuohch@lzu.edu.cn](mailto:zuohch@lzu.edu.cn) (H. Zuo).

methods have been proposed. These methods can be divided into five types: physical models, statistical models, spatial correlation models [6], artificial intelligence models, and hybrid methods [7].

Physical models use the meteorological and geographic parameters provided by the Numerical Weather Forecast System (NWP), such as air pressure, temperature, topographic information, surface roughness, and obstacles, for numerical prediction [8,9]. They are suitable for long-term prediction [10] and show poor performance in short-term wind speed prediction [11]. By contrast, statistical methods use historical data for wind speed forecasting which is conducive to short-term forecasting. Autoregressive (AR) [11], autoregressive moving average (ARMA) [12,13], autoregressive integral moving average (ARIMA) [14], Kalman filtering [15], persistence method [16], the traditional statistical model composed of Markov chain model [17], and intelligent methods [18,19] only use historical data to describe the potential correlation between the future performance of wind speed and historical data. Compared with the physical model, a higher short-term wind speed prediction accuracy is achieved. However, a disadvantage of the statistical model is that the inherent linear assumption makes it lack the nonlinear fitting ability [20], making it difficult to handle a time series of wind speed with nonlinear characteristics.

In the past few decades, with the rapid development of artificial intelligence (AI) methods that have been developed and applied in many fields, and in particular in wind speed prediction in order to improve prediction performance [21]. Generally, artificial intelligence methods have good self-learning and self-organization capabilities and can approximate nonlinear functions [22]. Therefore, much research has been conducted on the application of artificial intelligence methods in wind speed prediction. In particular, as one of the main technologies of AI, machine learning models such as the back propagation neural network (BPNN), the Elman neural network, the echo state network (ESN) and the extreme learning machine (ELM) [23] can deal well with the inherent characteristics of wind speed prediction [24,25]. The advantages of these methods have been examined in many studies [26,27].

However, traditional machine learning methods cannot extract feature information of time series from data in detail because they lack deep extraction capabilities [28,29]. Fortunately, in recent years, due to the advancement of computing power, deep learning methods have developed rapidly [30]. One of the advantages of deep learning is that it can automatically construct more complex features than other machine learning methods at each calculation step [31]. Moreover, deep learning has been proven to be effective in achieving high accuracy in wind speed prediction [32]. However, methods such as the deep confidence network (DBN) [2] and the stacked automatic encoder (SAE) [33] cannot extract data features that require higher time series information. Recently, recurrent neural networks (RNN) have been used to predict wind speed [34]. This network can receive information from the previous state to the current state by maintaining storage units in the hidden layer. Therefore, this kind of network is a typical deep neural networks (DNN) suitable for prediction of wind speed based on historical time series data [35,36].

According to our literature survey, although deep learning methods have achieved good results in the field of time series prediction, it is difficult for a single deep network model to adapt to all wind speed conditions. To further improve the prediction performance of deep networks, researchers continue to add new data processing techniques and optimization algorithms to improve the model, thus proposing many hybrid models that combine the advantages of different methodologies. The hybrid prediction model is considered to be the mainstream method, and the number of studies focusing on this method has been greatly increased [20,37,38]. Generally, the mixed model mainly includes signal

preprocessing algorithm, prediction algorithm and optimization algorithm. To reduce the nonstationarity of wind speed data, signal preprocessing methods, particularly decomposition methods, are widely used in wind speed sequence prediction models. First, the method uses signal decomposition technology to decompose the original wind speed into several components. Then, each component is predicted through an appropriate model. The final prediction result is the sum of the prediction results of each model. Among these signal decomposition techniques, wavelet transform (WT) and empirical mode decomposition (EMD) [39] have been widely used for wind speed prediction. In addition, numerous experts have proposed a series of improved decomposition methods such as ensemble EMD (EEMD) [40,41], the complete EEMD with adaptive noise (CEEMDAN) [42] and the improved CEEMDAN (ICEEMDAN) [43]. Based on the literature survey, the following conclusions can be drawn: the decomposition method can improve the predictor's recognition ability and greatly improve the prediction accuracy.

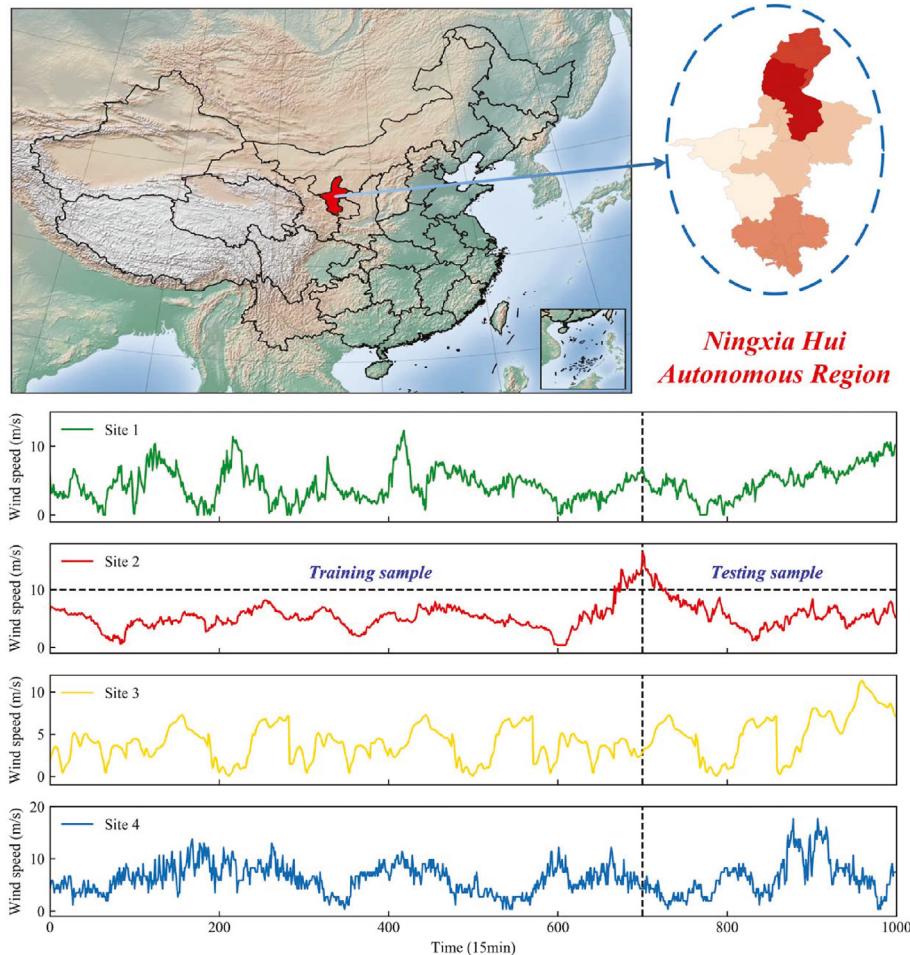
Based on the above considerations, this paper proposes a new hybrid model combining ICEEMDAN, RNN, ARIMA and error correction methods for short-term wind speed prediction. The model consists of decomposition module, prediction module and error correction module. First, the ICEEMDAN module can be used to convert nonlinear and non-stationary wind speed data into a series of relatively simple subseries. The RNN model is established as a prediction module to predict each subseries. Then, the predicted wind speed and error can be obtained. Then, the error is decomposed by ICEEMDAN, and the statistical model ARIMA is used to predict the error subsequence and obtain the prediction error. Finally, the final predicted wind speed can be obtained by summing the previously predicted wind speed and the current predicted error. Compared to the relevant research, the main contributions of this study are as follows. (1) A wind speed prediction system based on ICEEMDAN decomposition, RNN and error correction is proposed. (2) The prediction performance of the two networks of the proposed RNN and the traditional BPNN are compared in detail. (3) The prediction performances of the proposed system with and without error correction are compared using common evaluation criteria.

The rest of this paper is arranged as follows. In section 2, the wind speed data used in the experiment are briefly introduced. Section 3 describes the main process of the proposed hybrid model. The specific methodology used in proposed model, including ICEEMDAN decomposition, BPNN, two kinds of RNNs, ARIMA model, and error correction, is described in detail in section 4. At the same time, taking BPNN as an example, the detailed process of error decomposition, prediction and correction is analyzed. Section 5 describes the performance metric used in this study. In section 6, we discuss the prediction results of the proposed model and compare the results of different models. The last section draws some important conclusions based on the results of the previous section.

## 2. Experimental data

To prove the prediction effectiveness of the proposed model, we use four wind speed series collected from a wind farm in the Ningxia Hui Autonomous Region of China. Each series contains 1000 samples and is divided into training series and testing series. The first 700 samples of each site series are used for training and the rest are used for testing, as shown in Fig. 1. The height of the measured wind speed is 70 m, and the sampling interval is 15 min.

Moreover, some statistical indicators for the wind speed data of four sites, including maximum, minimum, mean, and standard deviation, are presented in Table 1.



**Fig. 1.** Original wind speed series from four sites.

**Table 1**  
Site details.

Site name	Max.(m/s)	Min.(m/s)	Mean(m/s)	Std.(m/s)
Site 1	12.29	0	4.4163	2.356
Site 2	16.5	0.4	5.4374	2.346
Site 3	11.31	0.08	4.1876	2.33
Site 4	17.71	0.37	6.2353	2.9934

### 3. Framework of hybrid model

The framework of the proposed hybrid model is shown in Fig. 2. The core of the model is to predict the error and to correct the previously forecast wind speed by using the forecasting error, to improve the accuracy of wind speed prediction.

In summary, the proposed framework is divided into the following steps:

- (1) Use ICEEMDAN to decompose original wind series into several subseries and residual.
- (2) Using artificial neural network to predict each subseries, the prediction series and error series are obtained.
- (3) Use ICEEMDAN again to decompose the error series into subseries, and use the ARIMA model to predict each subseries, finally, obtain the forecasting error series.

- (4) Use the forecasting error series to correct the forecasting series in the second step to obtain the final forecasting wind speed.

## 4. Methodology

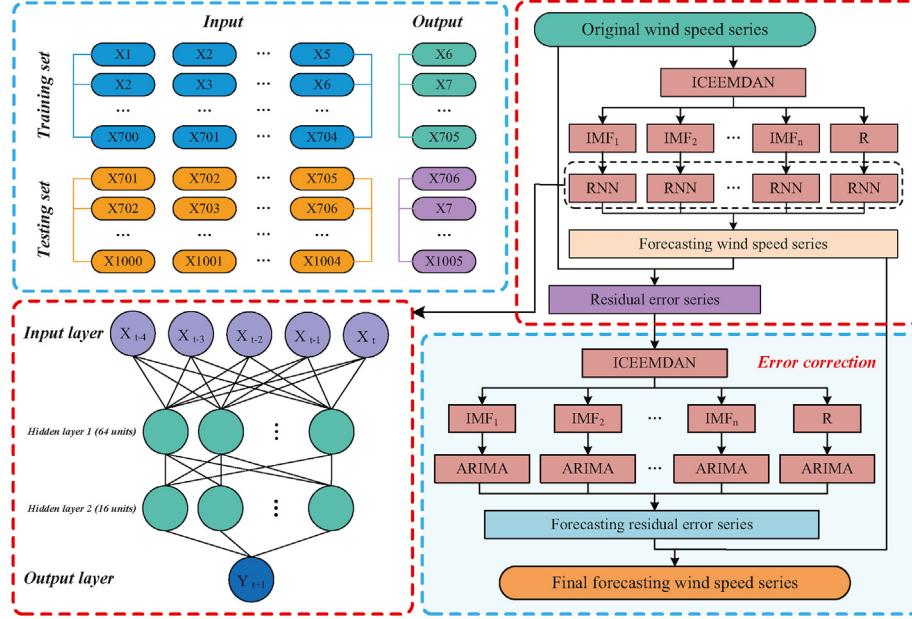
In this paper, the ICEEMDAN, ARIMA, RNN, and BPNN that are applied in the proposed hybrid model are introduced as follows.

### 4.1. ICEEMDAN

EMD [44], proposed by Huang et al., is an adaptive method for the analysis of nonstationary and nonlinear signals. EMD can decompose the original signal into the sum of amplitude and frequency modulation functions, called “intrinsic mode function” (IMF), and the final monotonic trend. In recent years, a novel technique called ICEEMDAN, proposed by Marcelo et al. [43] has been employed for reducing the noise and uncertainty of wind speed series in this study.

Given a composite signal  $x(t)$ ,  $t$  is the sampling sequence of the signal. Let  $E_k(\cdot)$  be the  $k$ th IMF obtained by EMD, and define  $M(\cdot)$  as the operator to calculate the local mean of the signal, then the ICEEMDAN algorithm is described as follows [45]:

**Step 1:** Calculate the local means of realizations using the EMD algorithm;



**Fig. 2.** Framework of the proposed ICEEMDAN-RNN-ICEEMDAN-ARIMA models.

$$x^i = x + \beta_0 E_1(w^i)$$

where  $\beta_0 = \varepsilon_0 \text{std}(x)/\text{std}(E_1(w^i))$ ,  $\text{std}(\cdot)$  is the operator of standard deviation, and  $\varepsilon_0$  is the reciprocal of the desired signal-to-noise ratio between the first added noise and the analyzed signal.

**Step 2:** Calculate the first residue  $R_1$ :

$$R_1 = M(x^i)$$

**Step 3:** Compute the first mode at the first stage ( $k = 1$ ) using the following formula:

$$d_1 = x - R_1$$

**Step 4:** Estimate the second residue as the average of local means of the realizations  $R_1 + \beta_1 E_2(w^i)$  and define the second mode:

$$d_2 = R_1 - R_2 = R_1 - M(R_1 + \beta_1 E_2(w^i))$$

**Step 5:** For  $k = 3, \dots, K$ , calculate the  $k_{th}$  residue:

$$R_k = M(R_{k-1} + \beta_{k-1} E_k(w^i))$$

$$\beta_k = \varepsilon_0 \text{std}(r_k), k \geq 1$$

**Step 6:** Compute the  $k_{th}$  mode:  $d_k = R_{k-1} - R_k$

$$R_k = M(R_{k-1} + \beta_{k-1} E_k(w^i))$$

**Step 7:** Go back to step 4 for next  $k$ .

Compared with other decomposition methods, using ICEEMDAN can not only reduce the noise in the mode but also decrease the residual spurious pattern problems caused by signal overlap, making the decomposition more accurate and thus providing an accurate reconstruction of the original signal [46].

For the original wind speed data of site 4, the IMF components

obtained by ICEEMDAN are shown in Fig. 3.

#### 4.2. Back-propagation neural network (BPNN)

The BPNN is a typical multilayer feedforward neural network in an artificial neural network that has a wide range of applications. It is based on the gradient descent method that can be compared with the preset expected output according to the actual input of each neuron during training. If it is not within the allowable error range, the weight will be readjusted by back-propagation until the training is completed. The process of this network training is divided into two steps, namely, updating and learning [47].

$$w_{ij}(t) = w_{ij}(t-1) - \Delta w_{ij}(t) \quad (1)$$

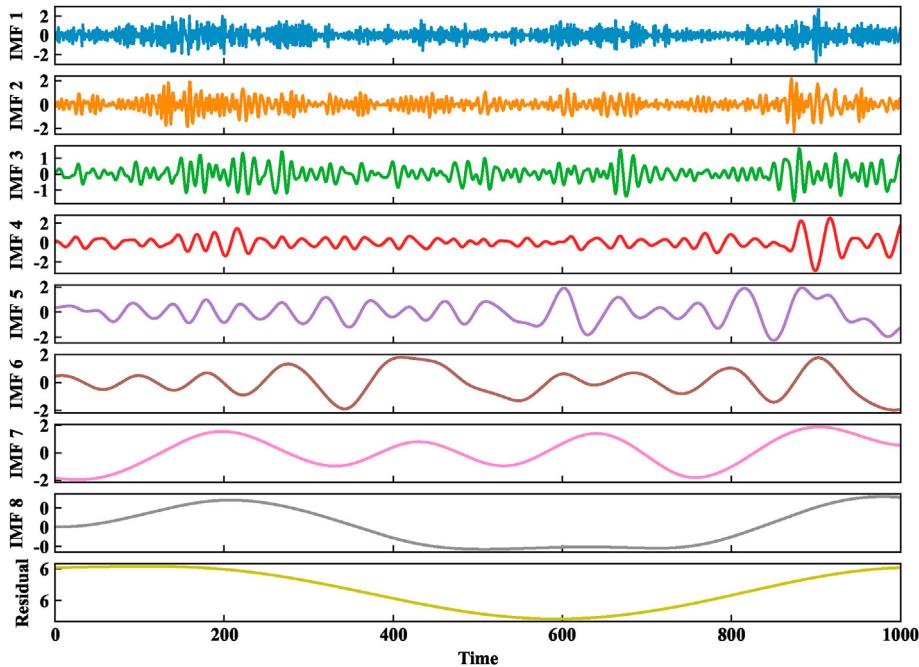
$$\Delta w_{ij}(t) = \eta \frac{\partial E}{\partial w_{ij}}(t-1) + \alpha \Delta w_{ij}(t-1) \quad (2)$$

where  $w$  is the connection weight of nodes  $i$  and  $j$ ,  $\eta$  is the learning rate,  $E$  is the gradient of the error function and  $\alpha \Delta w_{ij}(t-1)$  is the weight size step.

#### 4.3. Recurrent neural network (RNN)

In the traditional neural network, the nodes are connected between the layers, and there is no connection between the nodes in the same hidden layer. However, in RNN, the nodes in the same hidden layer are connected to each other, that is, the output of the current moment is not only related to the input of the current moment but also depends on the output of the previous hidden layer. Therefore, RNN can encode the prior information into the learning process of the current hidden layer, so that it can effectively learn time series data.

In recent years, RNN has achieved great success and been widely used in natural language processing (such as speech recognition, machine translation, etc.) because of its remarkable effect in dealing with time series data with long-term dependence. The



**Fig. 3.** Result of the original wind speed of Site 4 decomposed by ICEEMDAN.

main idea of RNN is the built-in loop structure, and the loop in RNN allows information to be transferred from the previous layer of the network to the next layer. Its chain nature shows that cyclic neural network is closely related to sequence problems, such as speech recognition and language modeling. In addition, RNN allows each value of the time step to share parameters, and the statistical strength of different time steps is shared at the adjacent time step, greatly enhancing its feature extraction ability. All of these inherent features enable RNN to outperform simple multilayer perceptrons and other deep learning architectures.

However, in practical application, when the interval between the previous information and the current forecasting position is large, the RNN cannot remember the previous information very well, and there exists the problem of gradient disappearance or gradient explosion, that is, when the gradient signal multiplies the weight matrix of neurons along the hidden layer, if the weight matrix is too small or too large, the gradient signal will eventually become so small that learning stops working, or too large that learning diverges. Therefore, the forecasting result is not ideal.

#### 4.4. Long short-term memory network (LSTM)

The LSTM network is a special RNN that was proposed by Hochreiter and Schmidhuber et al., in 1997 [48]. Its main idea is to introduce an adaptive gating mechanism that determines the extent to which the memory unit maintains the previous state while at the same time, it remembers the characteristics of the current input data, so that it is highly suitable for processing and predicting events with long intervals and delays in time series. Furthermore, the LSTM network replaces the neurons in the hidden layer of the RNN with the memory unit to realize the memory of the past information. Each memory unit contains one or more memory cells and three gate. As shown in Fig. 4, it consists of an input gate, an output gate and a forget gate in a unit. Among these, the “gate” of the long-term and short-term memory network is a special network structure for which the input is a vector and the output range is 0–1. When the output value is 0, no information is allowed

to pass through; when the output value is 1, all information is allowed to pass through. Therefore, it will selectively discard a part of the information, avoiding the problem of gradient disappearance and gradient explosion in RNN.

Let input  $x = (x_1, x_2, \dots, x_t)$  be the existing historical data and output  $y = (y_1, y_2, \dots, y_t)$  the data to be forecasting, then the calculation process of this network is given as follows [47].

$$\begin{aligned} i_t &= \sigma(W_{xi}X_t + W_{hi}h_{t-1} + b_i + W_{ci} \odot c_{t-1}) \\ f_t &= \sigma(W_{xf}X_t + W_{hf}h_{t-1} + b_f + W_{cf} \odot c_{t-1}) \\ y_t &= \tanh(W_{xy}X_t + W_{hy}h_{t-1} + b_y) \\ c_t &= f_t \odot c_{t-1} + i_t \odot y_t \\ o_t &= \sigma(W_{xo}X_t + W_{ho}h_{t-1} + b_o + W_{co} \odot c_t) \\ h_t &= o_t \odot \tanh(c_t) \end{aligned} \quad (3)$$

where  $i_t$ ,  $f_t$  and  $o_t$  are the input, forget and output gates, respectively.  $\sigma$  and  $\odot$  signify the logistic sigmoid function and element-wise multiplication, respectively.  $c_t$  is the current cell status value.  $b_*$  denotes the bias vectors. All of the weight matrices  $W_*$  are updated using the error back-propagation algorithm [49] according to the difference between the output value and the actual value.

#### 4.5. Gated recurrent unit (GRU)

GRU is another kind of RNN that was first proposed by Cho et al., in 2014 [50]. The typical structure of GRU cells is shown in Fig. 5. Each GRU unit has one less gate than the LSTM unit and consists of two gates: the reset gate  $r$  and the update gate  $u$ , so that the GRU is easier to calculate and implement. Similar to the LSTM unit, the hidden state at time  $t$  and the input time series value at time  $t$  are used to calculate the hidden state at time  $t - 1$ , and the function of the reset gate is similar to the forget gate of LSTM. The regression part and optimization method for GRU in this article are the same as those for LSTM.

The main process in a GRU unit can be described as follows.

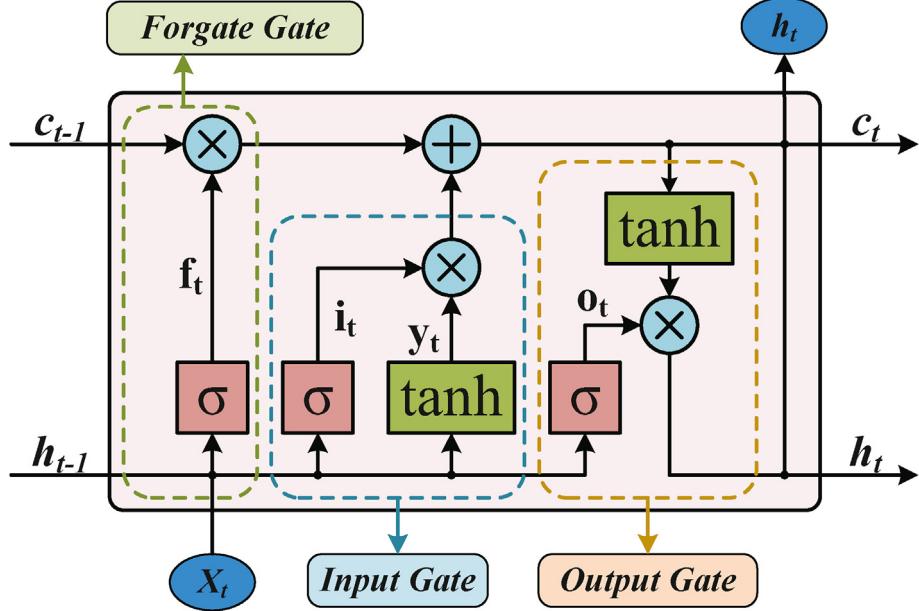


Fig. 4. Unit model of LSTM.

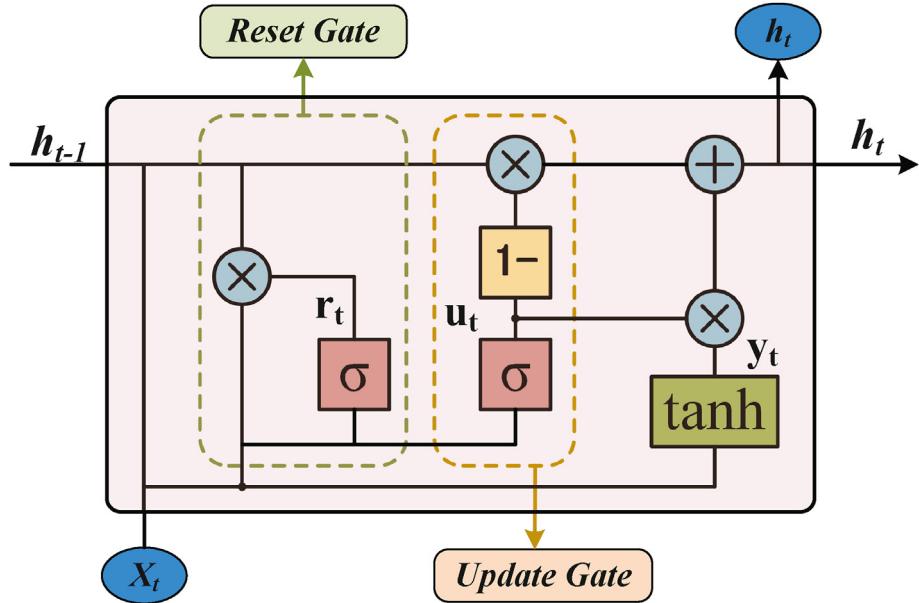


Fig. 5. Unit model of GRU.

$$\begin{aligned}
 r_t &= \sigma(W_{xr}x_t + W_{hr}h_{t-1} + b_r) \\
 u_t &= \sigma(W_{xu}x_t + W_{hu}h_{t-1} + b_u) \\
 y_t &= \tanh(W_{xh}x_t + W_{hy}(r_t \odot h_{t-1}) + b_y) \\
 h_t &= (1 - u_t) \odot h_{t-1} + u_t \odot y_t
 \end{aligned} \tag{4}$$

where  $h_{t-1}$  is the hidden layer status at the previous moment, and  $h_t$  is the hidden layer information at the current moment.  $u_t$  and  $r_t$  are the update and reset gates of the GRU, respectively, and the candidate hidden layer  $y_t$  is calculated by  $r_t$ , thereby measuring how much hidden layer information is retained at the previous moment.  $u_t$  is used to calculate how much candidate hidden layer  $y_t$  needs to be added and finally obtain the output  $h_t$  at the current

moment.

Next, three neural networks, namely, BPNN, LSTM, and GRU are used to predict the IMF components after ICEEMDAN decomposition. Considering the efficiency of training, we map the wind speed data to between 0 and 1 prior to training, and the formula is

$$x_{normal} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{5}$$

where  $x$  are the original data,  $x_{normal}$  are the normalized data,  $x_{max}$  and  $x_{min}$  are the maximum and minimum of the original datasets, respectively.

After the training is completed, the forecasting data can be obtained by using the trained model, and the actual forecasting value

can be obtained by its anti-normalization through Eq. (5). Taking the wind speed data at Site 4 as an example, the prediction results of each IMF are shown in Fig. 6(a). By summing the forecasting IMF components, the wind speeds forecasting by the three networks are shown in Fig. 6(b). At the same time, we have obtained the training error series of the three kinds of networks shown in Fig. 6(c).

#### 4.6. ARIMA model

The ARIMA is a mathematical method for the analysis of a stationary time series [51]. Its core idea is that some time series data change with time, which appears to have an obvious uncertainty; however, from the point of view of overall change, there is a certain change rule. The model is used to describe this change rule approximately and to predict the data at some future time. With the aid of this model, the optimal prediction under the minimum variance can be obtained by analysis and studying the time series data. Therefore, this model is used to predict the error wind speed series.

In the ARIMA model, the predicted values of the variables are assumed to be linear functions of several past observations and random errors. In other words, the process of predicting future time series has the following form:

$$y_t = \sum_{i=1}^p \beta_i y_{t-i} + \sum_{j=1}^q \alpha_j e_{t-j} + \varepsilon_t$$

where  $p$  is the order of the autoregressive part,  $\beta_i$  is the autoregressive parameter,  $q$  is the order of the moving average part,  $\alpha_j$  is the moving average parameter, and  $\varepsilon_t$  is the error term at time  $t$ .

The main task of ARIMA model building is to determine the appropriate model order. To determine the best values of  $p$  and  $q$ ,

we use the Akaike information criterion (AIC) [52] that is defined as follows.

$$AIC = \log V + 2d/N$$

where  $V$  is the loss function,  $d$  is the number of estimated parameters, and  $N$  is the number of values in the estimation data set.

The loss function  $V$  is defined by the following equation:

$$V = \left( \sum_1^N \varepsilon(t, \hat{\theta}_N) (\varepsilon(t, \hat{\theta}_N))^T / N \right)$$

where  $\hat{\theta}_N$  are the estimated parameters.

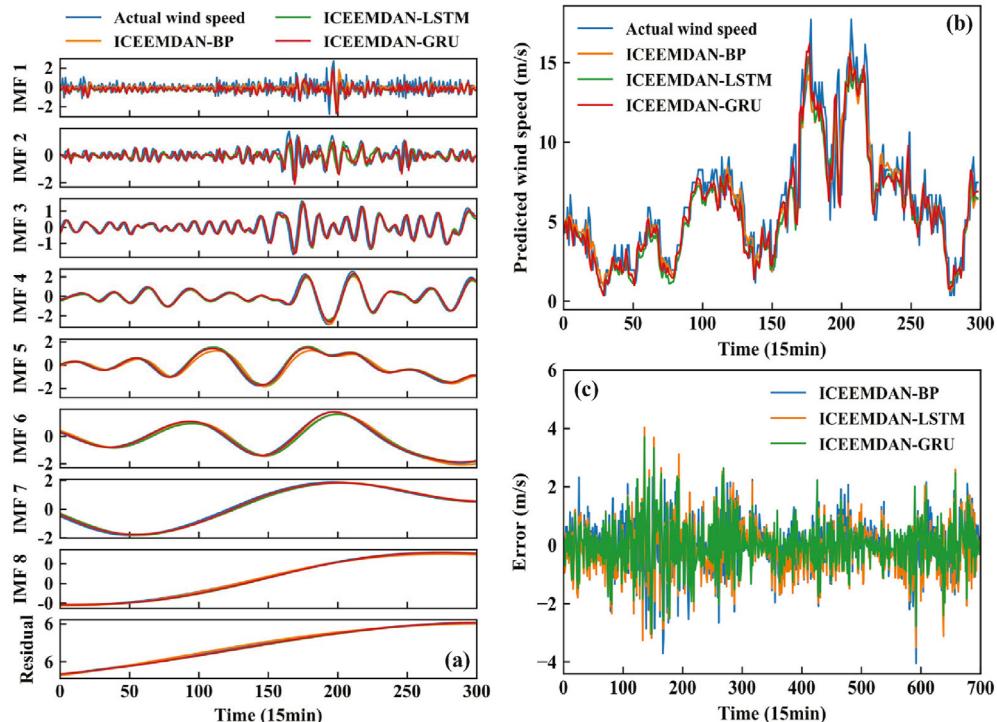
#### 4.7. Error correction

To further improve the accuracy of wind speed forecasting, we use the error correction method to solve this problem. The specific correction steps are shown in the shadow section in Fig. 2. First, after RNN neural network training, the error series  $E(t)$  of the training set is obtained from the original wind speed data, and this can be expressed as

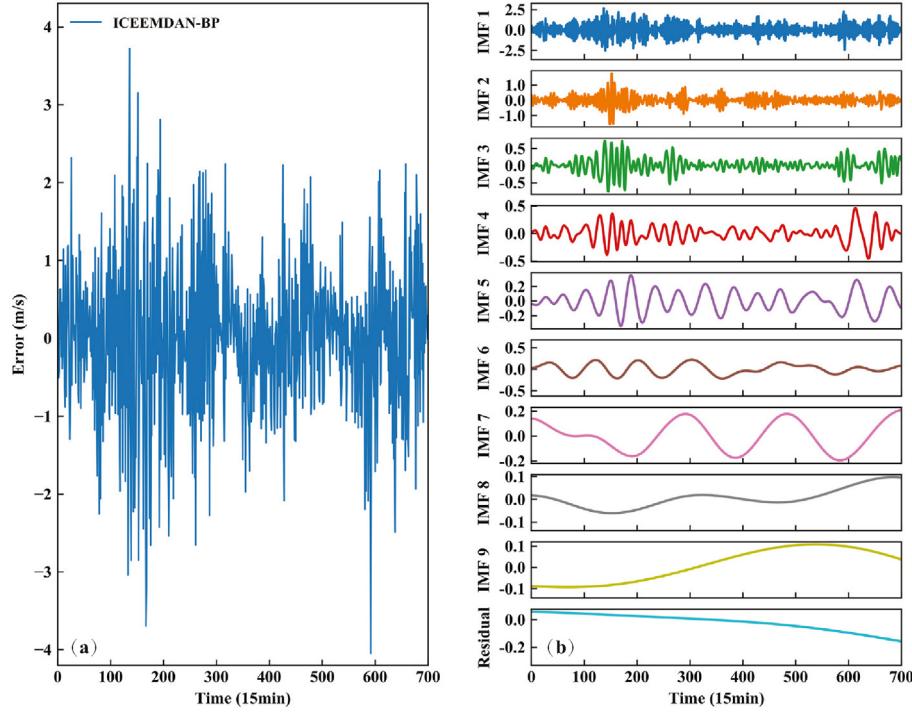
$$E(t) = W_{RNN}(t) - W_{act}(t) \quad (6)$$

where  $W_{RNN}(t)$  is the wind speed forecasting during the training of the RNN network and  $W_{act}(t)$  is the actual wind speed. The training error series obtained by the BPNN and two kinds of RNN are shown in Fig. 6(c).

Similar to the previous forecasting of wind speed, we also decompose the error series by ICEEMDAN, and use the ARIMA model to predict the decomposed IMF components, and then accumulate the prediction results of each IMF to obtain the final error prediction series. Here, taking Site 4, BPNN as an example, the



**Fig. 6.** (a) Original IMF components and predicted IMFs at Site 4, (b) Original wind speed and predicted wind speed at Site 4. (c) Errors obtained after prediction at Site 4.



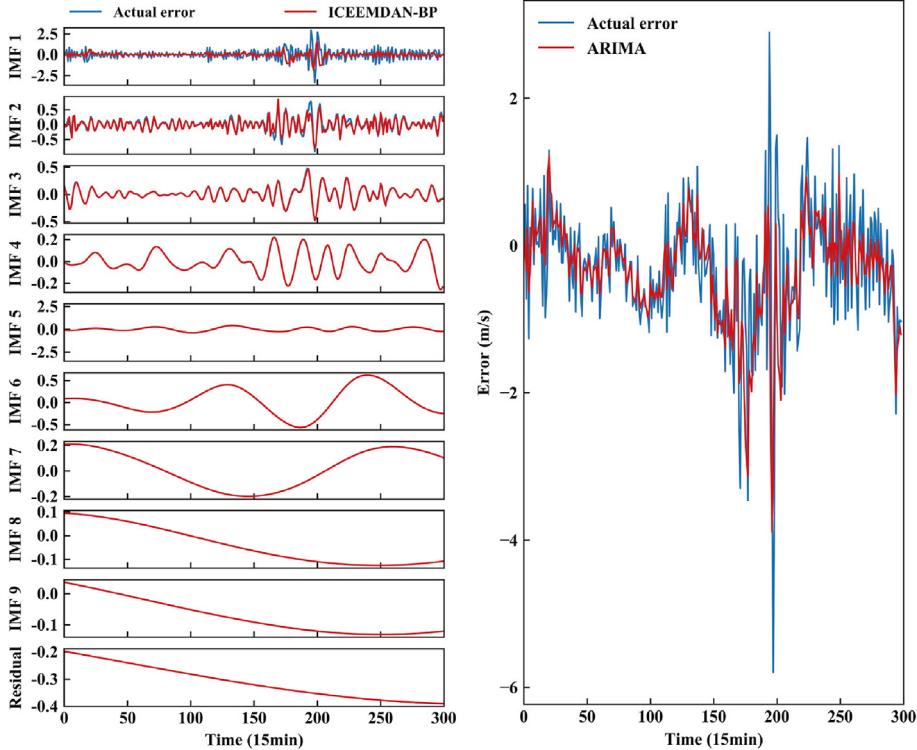
**Fig. 7.** (a) Error series obtained by BPNN at Site 4. (b) BPNN's error IMFs obtained by ICEEMDAN at Site 4.

training error is shown in Fig. 7(a) and the decomposed IMFs is shown in Fig. 7(b). The IMFs forecasting by the ARIMA model and the total prediction error series are shown in Fig. 8.

After the total error is obtained by using the ARIMA model, the final forecasting wind speed can be obtained as follows

$$\widehat{W}_{fin}(t) = \widehat{W}_{RNN}(t) + \widehat{E}(t) \quad (7)$$

where  $\widehat{W}_{RNN}(t)$  is the RNN for wind speed and  $\widehat{E}(t)$  is the ARIMA for error. Finally, the final forecasting wind speed of three neural networks with error correction in Site 4 is shown in Fig. 9.



**Fig. 8.** Original error and ARIMA prediction error (left), and total prediction error (right) at Site 4.

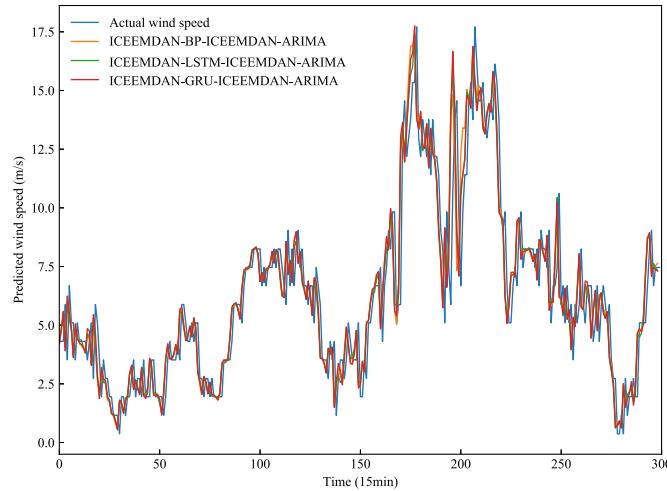


Fig. 9. Forecasting results of with error correction model for Site 4.

## 5. Performance metrics

To understand the effect of the model in practical application more intuitively and to correctly evaluate the performance of the proposed model, some commonly used indicators are selected in this paper, including the mean absolute error (MAE), root mean square error (RMSE), mean absolute percentage error (MAPE) and the sum of error squares (SSE). These four performance metrics are defined as follows:

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2}$$

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100\%$$

$$SSE = \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

where  $y_i$  and  $\hat{y}_i$  are the actual and forecast wind speed at time t, respectively, and N is the number of the samples.

MAE, RMSE, MAPE and SSE are commonly used to measure the error in time series analysis. Among these, MAE is the average value of the absolute error between the forecasting value and the actual observed value that can accurately reflect the actual error; RMSE represents the sample standard deviation of the difference between the forecasting value and the actual observed value that can reflect well the precision of prediction. Because a larger error has a disproportionate influence on MAE and RMSE, they are sensitive to abnormal values. MAPE compensates for this shortcoming. It expresses the accuracy as a percentage by dividing the absolute error by its corresponding actual value. Finally, SSE represents the total error of the model. For problems, smaller values of MAE, RMSE, MARE and SSE indicate better performance.

In addition, to compare the wind speed forecasting performance of the different models, the promoting percentages of the MAE ( $P_{MAE}$ ), the promoting percentages of the RMSE ( $P_{RMSE}$ ), the promoting percentages of the MAPE ( $P_{MAPE}$ ), and the promoting percentages of the SSE ( $P_{SSE}$ ) are introduced in this study. They are defined as follows [53]:

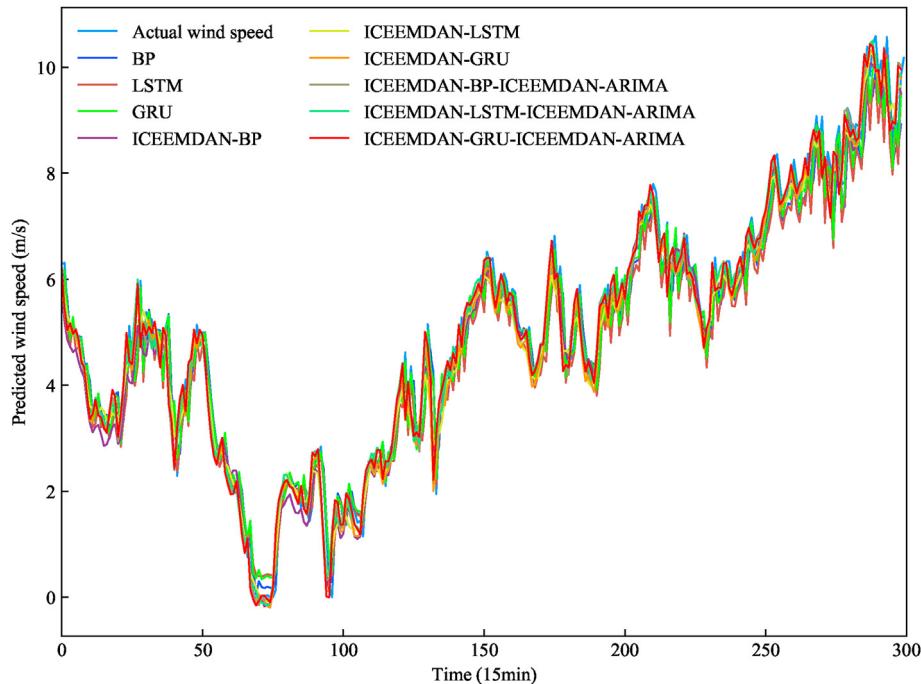
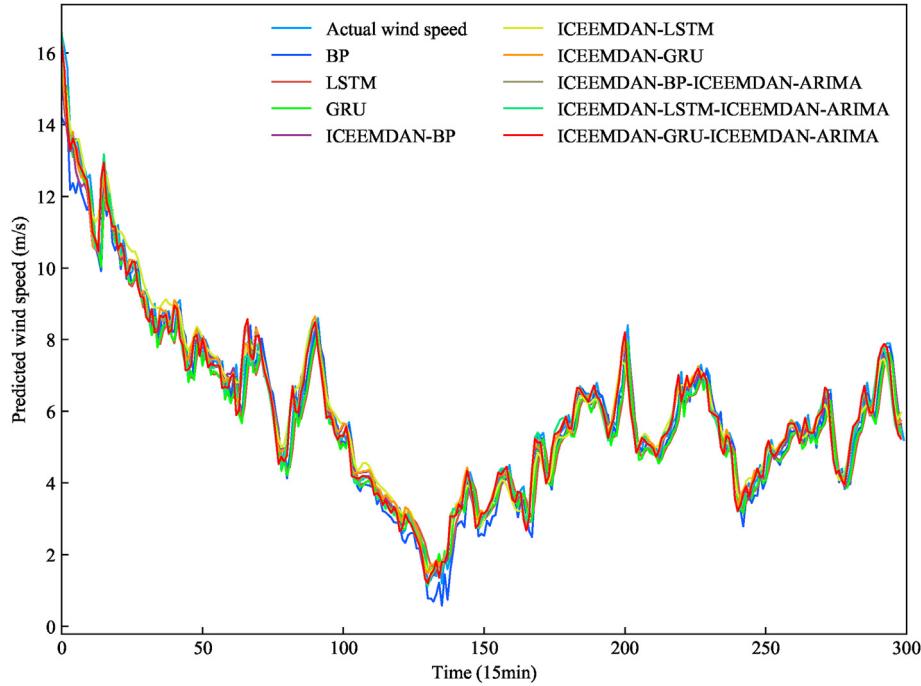
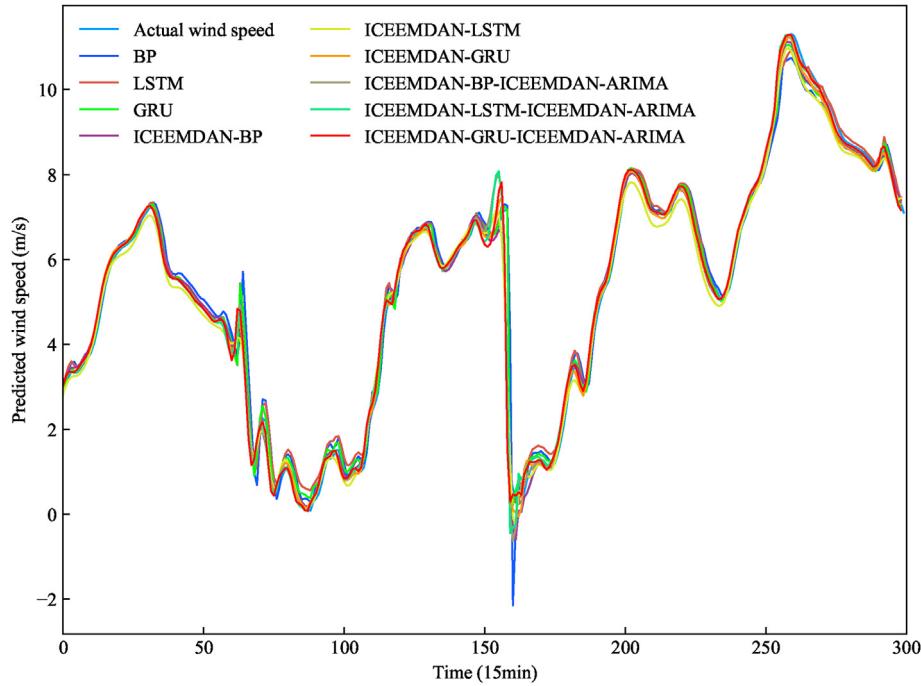


Fig. 10. Forecasting results of the models without error correction and with error correction for Site 1.



**Fig. 11.** Forecasting results of the models without error correction and with error correction for Site 2.



**Fig. 12.** Forecasting results of the models without error correction and with error correction for Site 3.

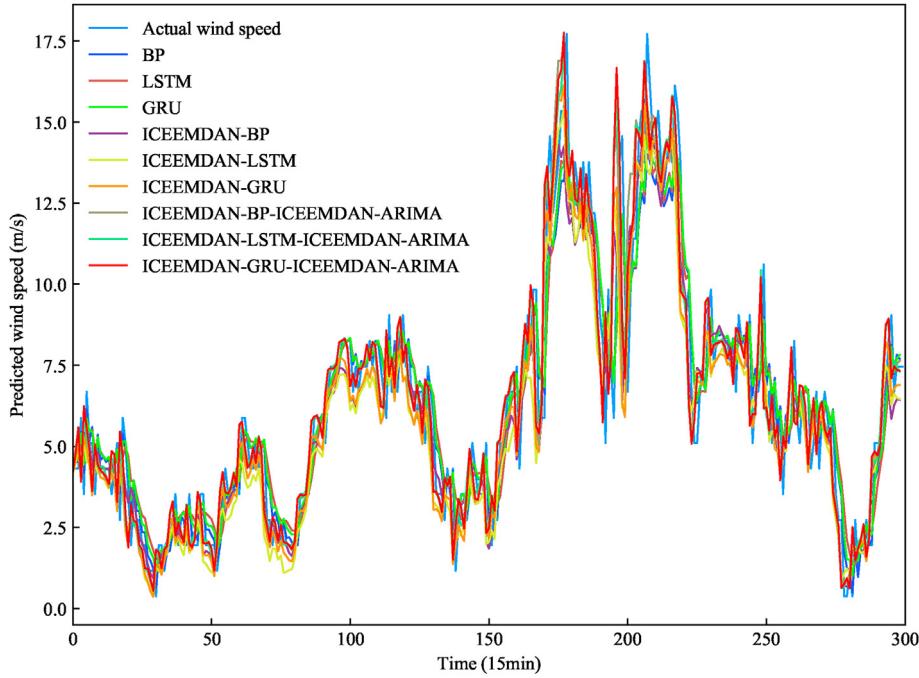
$$\begin{aligned}
 P_{MAE} &= |MAE_m - MAE_n| / MAE_n \\
 P_{RMSE} &= |RMSE_m - RMSE_n| / RMSE_n \\
 P_{MAPE} &= |MAPE_m - MAPE_n| / MAPE_n \\
 P_{SSE} &= |SSE_m - SSE_n| / SSE_n
 \end{aligned}$$

where  $m$  and  $n$  represent arbitrary two different models. These quantities indicate the degree of prediction performance improvement of one model relative to another model. A larger

value indicates a better prediction performance of model  $m$  than model  $n$ .

## 6. Results and discussion

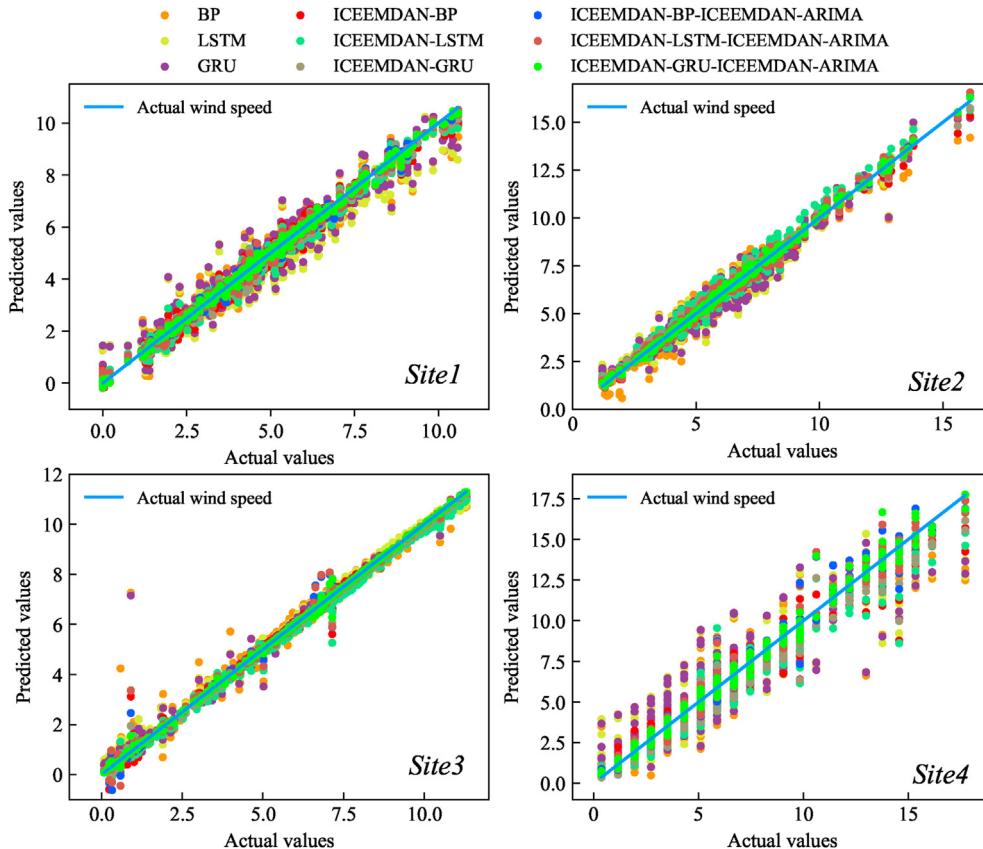
To improve the reliability of the model, each experiment is repeated three times, and the average value is regarded as the forecasting performance of each model. All of the simulations operate are carried out using python on Windows 10 with a 3.0 GHz



**Fig. 13.** Forecasting results of the models without error correction and with error correction for Site 4.

Intel Core i7 9700 CPU and 16 GB RAM. In addition, BPNN and RNN are developed using TensorFlow, an open source deep learning framework provided by Google [54].

Nine groups of experiments were carried out on the wind speed data with each site. BPNN, LSTM and GRU were used to carry out the first three groups of experiments. The second three groups



**Fig. 14.** Scatter plots of the results of the nine models for Site 1–4.

**Table 2**

Comparison of the forecasting performance of different models.

Model	Evaluation Criteria	Site 1	Site 2	Site 3	Site 4
BP	MAE (m/s)	0.5019	0.4951	0.2793	1.1535
	RMSE (m/s)	0.6348	0.6313	0.5651	1.5572
	MAPE (%)	15.556	9.583	19.473	28.167
	SSE (m/s)	120.4953	119.1655	95.5039	725.1024
LSTM	MAE (m/s)	0.5419	0.4767	0.2087	1.1668
	RMSE (m/s)	0.6858	0.6148	0.4552	1.5656
	MAPE (%)	18.596	9.101	16.971	32.042
	SSE (m/s)	140.66	113.025	61.968	732.9528
GRU	MAE (m/s)	0.502	0.44523	0.1701	1.1381
	RMSE (m/s)	0.6284	0.565	0.426	1.5329
	MAPE (%)	17.657	8.341	13.119	29.895
	SSE (m/s)	118.0771	95.6865	54.276	702.595
ICEEMDAN-BP	MAE (m/s)	0.2718	0.1945	0.1199	0.6672
	RMSE (m/s)	0.3372	0.2649	0.2244	0.9232
	MAPE (%)	9.3	3.52	8.24	13.17
	SSE (m/s)	34.03	19.38	15.1	255.7
ICEEMDAN-LSTM	MAE (m/s)	0.2619	0.3153	0.2131	0.8088
	RMSE (m/s)	0.3306	0.399	0.2952	1.0671
	MAPE (%)	7.85	5.96	8.62	16.24
	SSE (m/s)	32.79	46.41	26.1	341.49
ICEEMDAN-GRU	MAE (m/s)	0.1914	0.1727	0.0836	0.6007
	RMSE (m/s)	0.2412	0.2246	0.1349	0.7932
	MAPE (%)	5.58	3.52	4.76	11.96
	SSE (m/s)	17.45	14.57	5.45	188.62
ICEEMDAN-BP-ICEEMDAN-ARIMA	MAE (m/s)	0.1544	0.1055	0.0617	0.3902
	RMSE (m/s)	0.1964	0.1364	0.1851	0.5894
	MAPE (%)	4.87	1.98	5.06	8.3
	SSE (m/s)	11.56	5.55	10.27	104.14
ICEEMDAN-LSTM-ICEEMDAN-ARIMA	MAE (m/s)	0.1544	0.1798	0.0758	0.356
	RMSE (m/s)	0.1977	0.2252	0.2223	0.508
	MAPE (%)	5.77	3.39	5.76	7.66
	SSE (m/s)	11.72	15.96	14.82	77.39
ICEEMDAN-GRU-ICEEMDAN-ARIMA	MAE (m/s)	<b>0.1095</b>	<b>0.096</b>	<b>0.0381</b>	<b>0.2849</b>
	RMSE (m/s)	<b>0.1379</b>	<b>0.124</b>	<b>0.0991</b>	<b>0.4362</b>
	MAPE (%)	<b>3.1</b>	<b>1.84</b>	<b>2.34</b>	<b>6.14</b>
	SSE (m/s)	<b>5.7</b>	<b>4.61</b>	<b>2.94</b>	<b>57.03</b>

**Note:** The values in bold are the best values.

without error correction are called ICEEMD-BP, ICEEMD-LSTM, and ICEEMD-GRU, respectively. Finally, the last three groups are with error correction, and the name ends with ARIMA.

Next, the forecasting results of all of the experimental models will be discussed and analyzed in detail. The results of each site are shown in Figs. 10–13, and the scatter plot of the results is shown in Fig. 14. Table 2 lists the four evaluation criteria introduced above for all of the models used on the four site series, namely, MAE(m/s), RMSE(m/s), MAPE(%), SSE(m/s), and the graphical results are shown in Figs. 15–18, respectively.

By comparison, Figs. 15–18 and Table 2 show the following:

(a) For Site 1, prior to error correction, the forecasting result of BPNN is not ideal. The SSE of BPNN and LSTM network forecasting results are 34.03 m/s and 32.79 m/s, respectively. However, after error correction, the forecasting result is slightly better than the forecasting result of the LSTM network. The SSE is 11.56 m/s and 11.72 m/s, respectively.

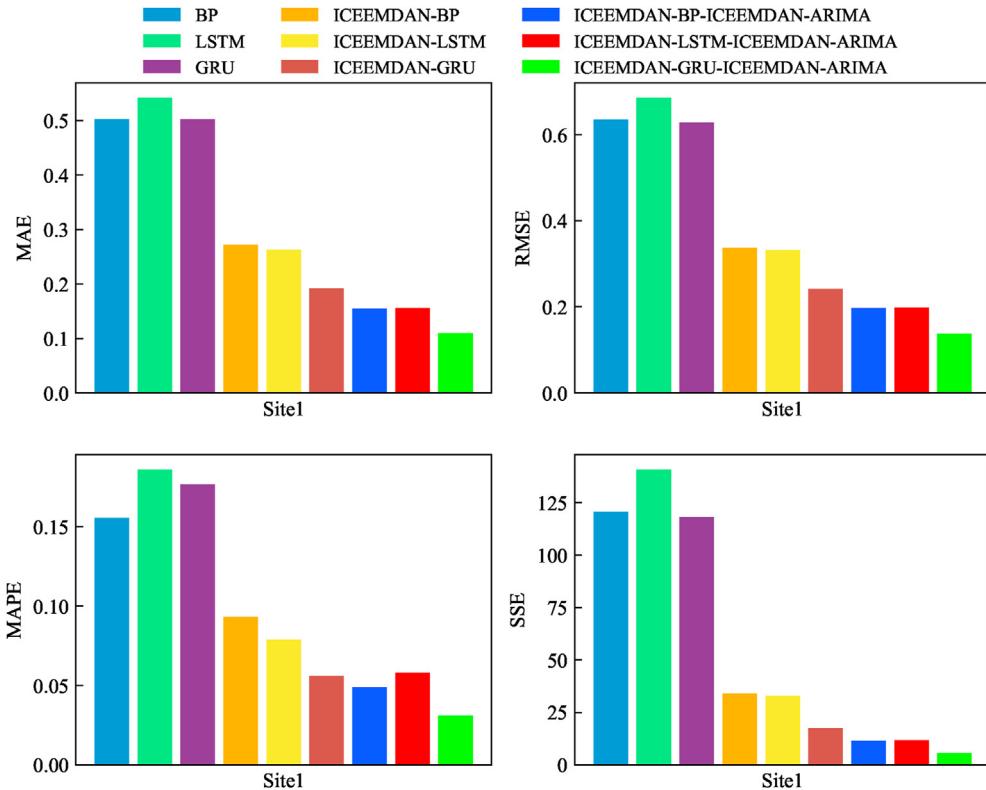
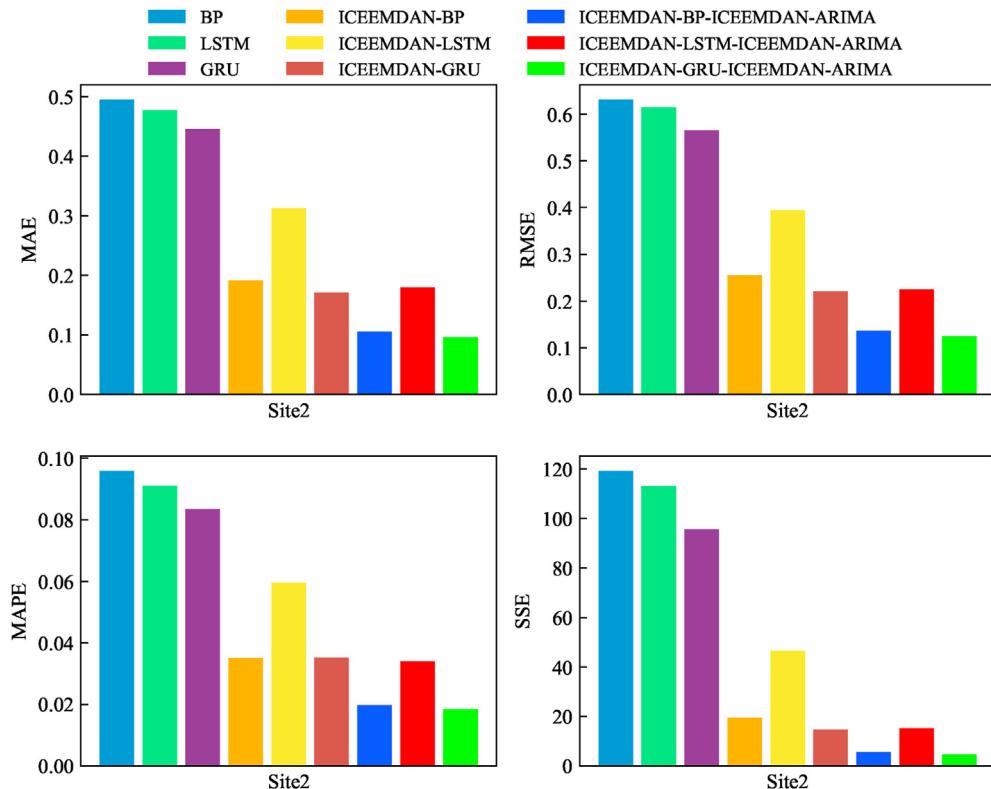
(b) In contrast to Site 1, for Site 4, prior to the error correction, the forecasting result of a BPNN is superior to the LSTM network forecasting result, but after error correction, the situation is reversed, and the result of the LSTM network is better than that of the BPNN.

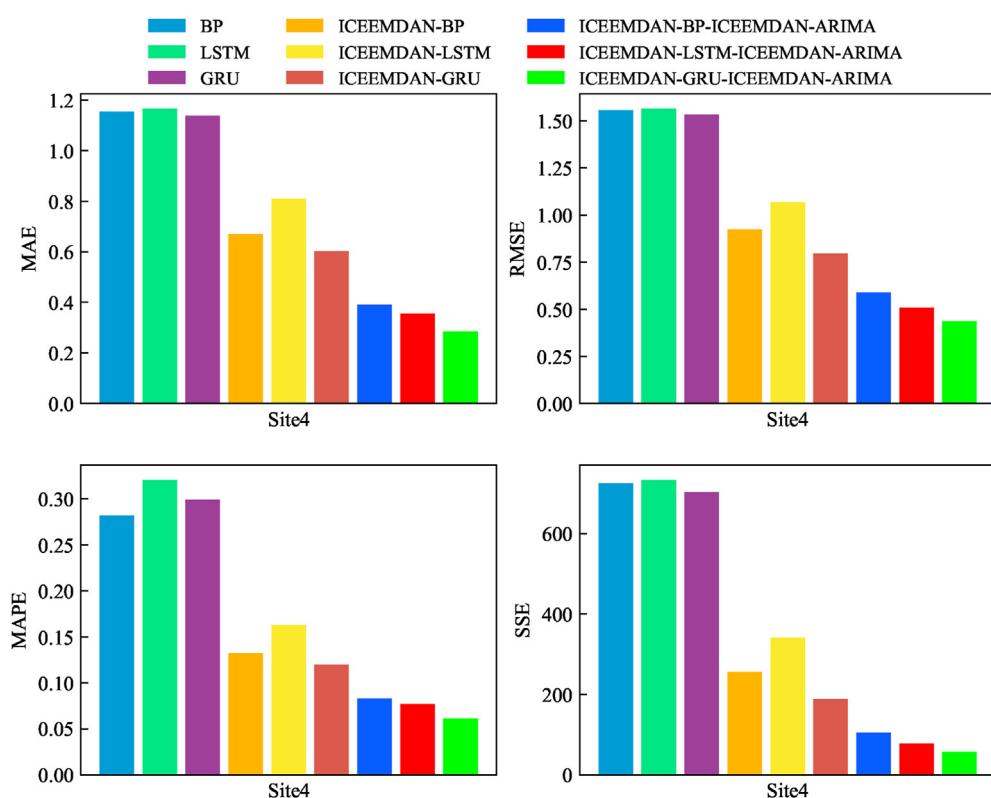
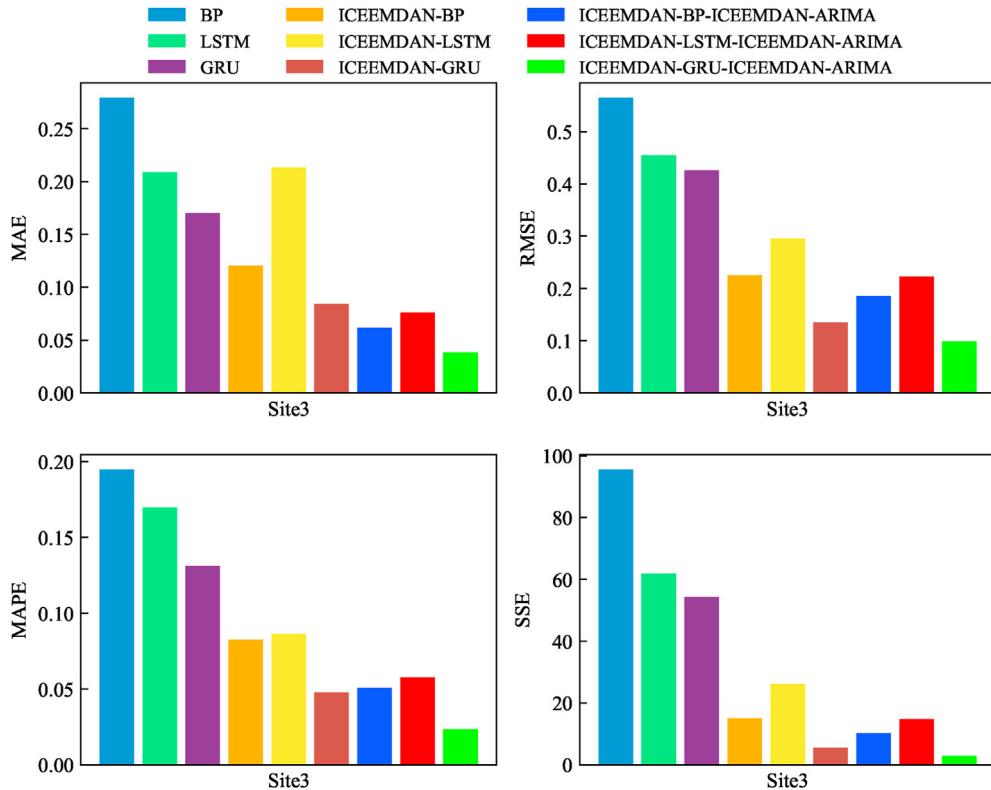
(c) For Sites 2 and 3, before and after the error correction, the performance of the results is consistent, that is, the performance of the network is the best, and the performance of BPNN is intermediate between those of the two RNN networks.

(d) Overall, the forecasting performance of the GRU network is the best both before and after correction. In addition, for all networks, the forecasting result after error correction is higher than the forecasting result without error correction.

To obtain the difference in prediction performance before the model more intuitively, we use the aforementioned promoting percentage to quantify the prediction performance between the models. As a matter of convenience, we use NN to represent any neural network in BP, LSTM, and GRU. Table 3 shows a comparison of the results obtained using the hybrid ICEEMD-NN model and a single NN model. Table 4 provides the results comparing the hybrid ICEEMD-NN–ICEEMDAN-ARIMA model and a single NN model. A comparison between the hybrid ICEEMD-NN–ICEEMDAN-ARIMA model and the hybrid ICEEMD-NN model is shown in Table 5. It is found that:

(a) The hybrid model based on GRU improves the prediction performance most obviously. The SSE promotion percentages of the single GRU model by the Predicted ICEEMDAN-GRU model are 85.217%, 84.771%, 89.943%, and 73.153%; the SSE promotion percentages of the single LSTM model by the Predicted ICEEMDAN-LSTM model are 76.687%, 58.936%, 57.868%, and 53.408%; and the SSE promotion percentages of the single BPNN model by the Predicted ICEEMDAN-BP model are 71.758%, 83.721%, 84.188%, and 64.735%. Furthermore, the SSE promotion percentage of the single GRU model by the Predicted ICEEMDAN-GRU-ICEEMDAN-ARIMA model are 95.168%, 95.181%, 94.572%, and 91.882%;

**Fig. 15.** Forecasting performance of different models for Site 1.**Fig. 16.** Forecasting performance of different models for Site 2.



**Table 3**

Promotion percentage of the hybrid ICEEMDAN-NN model of the wind speed series at each site to single NN model.

NN	Indexes	Site 1	Site 2	Site 3	Site 4
BP	$P_{MAE}$ (%)	45.865	61.459	57.002	41.974
	$P_{RMSE}$ (%)	46.857	59.653	60.236	40.616
	$P_{MAPE}$ (%)	40.102	63.37	57.58	53.079
	$P_{SSE}$ (%)	71.758	83.721	84.188	64.735
LSTM	$P_{MAE}$ (%)	51.558	34.454	2.137	30.537
	$P_{RMSE}$ (%)	51.717	35.919	35.091	31.742
	$P_{MAPE}$ (%)	57.555	34.605	49.138	49.219
GRU	$P_{SSE}$ (%)	76.687	58.936	57.868	53.408
	$P_{MAE}$ (%)	61.775	61.648	50.688	47.138
	$P_{RMSE}$ (%)	61.552	60.975	68.287	48.185
	$P_{MAPE}$ (%)	68.288	57.853	63.618	59.925
	$P_{SSE}$ (%)	85.217	84.771	89.943	73.153

**Table 4**

Promotion percentage of the hybrid ICEEMDAN-NN-ICEEMDAN-ARIMA model of the wind speed series at each site to single NN model.

NN	Indexes	Site 1	Site 2	Site 3	Site 4
BP	$P_{MAE}$ (%)	69.185	78.71	77.859	66.134
	$P_{RMSE}$ (%)	69.022	78.402	67.194	62.101
	$P_{MAPE}$ (%)	68.541	79.344	73.964	70.512
	$P_{SSE}$ (%)	90.404	94.335	89.238	85.637
LSTM	$P_{MAE}$ (%)	71.265	62.321	63.619	69.446
	$P_{RMSE}$ (%)	71.134	63.364	51.095	67.506
	$P_{MAPE}$ (%)	68.803	62.625	65.987	76.067
	$P_{SSE}$ (%)	91.668	86.578	76.083	89.441
GRU	$P_{MAE}$ (%)	78.145	78.397	77.557	74.95
	$P_{RMSE}$ (%)	78.018	78.048	76.702	71.507
	$P_{MAPE}$ (%)	82.347	77.914	82.127	79.441
	$P_{SSE}$ (%)	95.168	95.181	94.572	91.882

**Table 5**

Promotion percentage of the hybrid ICEEMDAN-NN-ICEEMDAN-ARIMA model of the wind speed series at each site to ICEEMDAN-NN model.

NN	Indexes	Site 1	Site 2	Site 3	Site 4
BP	$P_{MAE}$ (%)	23.32	17.25	20.857	24.16
	$P_{RMSE}$ (%)	22.166	18.75	6.958	21.486
	$P_{MAPE}$ (%)	28.44	15.974	16.385	17.434
	$P_{SSE}$ (%)	18.646	11.614	5.049	20.902
LSTM	$P_{MAE}$ (%)	19.707	27.867	65.756	38.908
	$P_{RMSE}$ (%)	19.417	27.446	16.004	35.764
	$P_{MAPE}$ (%)	11.248	28.02	16.85	26.847
	$P_{SSE}$ (%)	14.98	27.642	18.215	36.033
GRU	$P_{MAE}$ (%)	16.37	16.749	26.869	27.813
	$P_{RMSE}$ (%)	16.466	17.072	8.415	23.322
	$P_{MAPE}$ (%)	14.059	20.061	18.509	19.516
	$P_{SSE}$ (%)	9.95	10.41	4.629	18.729

the SSE promotion percentage of the single LSTM model by the Predicted ICEEMDAN-LSTM-ICEEMDAN-ARIAM model are 91.668%, 86.578%, 76.083%, and 89.441%; and the SSE promotion percentage of the single BPNN model by the Predicted ICEEMDAN-BP-ICEEMDAN-ARIAM model are 90.404%, 94.335%, 89.238%, and 85.637%. It is observed that the ICEEMDAN-GRU-ICEEMDAN-ARIMA structure greatly improves the accuracy of prediction.

(b) Regardless of which NN is used, the ICEEMDAN-NN-ICEEMDAN-ARIMA model with error decomposition correction always shows the best performance. Taking Site 4 as an example, after BP, LSTM, and GRU network errors are corrected, compared to before correction, the promotion percentages of SSE at Site 4 are 20.902%, 36.033% and 18.729%, respectively. The same conclusion can be

obtained for other sites. This shows that this hybrid ICEEMDAN-NN-ICEEMDAN-ARIMA has a certain universality and can in fact improve the accuracy of wind speed prediction.

## 7. Conclusions

To build an accurate short-term wind speed model, a new combination framework combining ICEEMDAN, BPNN, LSTM, GRU, and ARIMA is developed, and an error correction method based on ICEEMDAN-ARIMA is adopted. First, the collected wind speed time series is decomposed into several IMFs using ICEEMDAN, then the decomposed subseries are forecast separately by BPNN, LSTM, GRU, and the forecasting results are summarized. Second, according to the forecasting wind speed and the actual wind speed, the error wind speed series is obtained, and the error wind speed series is decomposed by ICEEMDAN again. Then, the decomposed error IMF series is forecast using the ARIMA model, and the forecasting error series is obtained after summing. Finally, the final result is obtained by summing the forecasting wind speed and the forecasting error. In the experimental phase, nine comparison models were implemented on four data sets collected from different sites, including three models without decomposition and error correction, three models with decomposition and no error correction, and three models with decomposition and error correction. According to the detailed results and the corresponding comprehensive analysis, the following conclusions can be drawn: (1) the hybrid model based on the GRU network shows better prediction performance than the hybrid model based on LSTM and BPNN, and (2) the error decomposition correction method can significantly improve the accuracy of prediction.

## Author contribution

Jikai Duan: Conceptualization, Methodology, Software, Validation, Writing Original draft preparation and Writing Reviewing. Hongchao Zuo: Supervision and Editing. Yulong Bai: Investigation and Editing. Jizheng Duan: Software. Mingheng Chang: Visualization. Bolong Chen: Data curation.

## Declaration of competing interest

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

## Acknowledgements

This research is funded by the National Natural Science Foundation of China project under grant number 41875009 and 41861047.

## References

- [1] Mi X, Zhao S. Wind speed prediction based on singular spectrum analysis and neural network structural learning. Energy Convers Manag 2020;216:112956.
- [2] Wang H, Wang G, Li G, Peng J, Liu Y. Deep belief network based deterministic and probabilistic wind speed forecasting approach. Appl Energy 2016;182: 80–93.
- [3] GWEC. The global wind report. [https://gwec.net/global-wind-report-2019/](https://gwec.net/global-wind-report-2019;); 2019.
- [4] Zhang Z-S, Sun Y-Z, Cheng L. Potential of trading wind power as regulation services in the California short-term electricity market. Energy Pol 2013;59: 885–97.
- [5] Wang J, Hu J. A robust combination approach for short-term wind speed forecasting and analysis—combination of the arima (autoregressive integrated moving average), elm (extreme learning machine), svm (support vector

- machine) and lssvm (least square svm) forecasts using a gpr (Gaussian process regression) model. *Energy* 2015;93:41–56.
- [6] Song J, Wang J, Lu H. A novel combined model based on advanced optimization algorithm for short-term wind speed forecasting. *Appl Energy* 2018;215:643–58.
- [7] Wu C, Wang J, Chen X, Du P, Yang W. A novel hybrid system based on multi-objective optimization for wind speed forecasting. *Renew Energy* 2020;146:149–65.
- [8] Sharma P, Bhatti T. A review on electrochemical double-layer capacitors. *Energy Convers Manag* 2010;51(12):2901–12.
- [9] Jiang P, Ma X. A hybrid forecasting approach applied in the electrical power system based on data preprocessing, optimization and artificial intelligence algorithms. *Appl Math Model* 2016;40(23–24):10631–49.
- [10] Naik J, Satapathy P, Dash P. Short-term wind speed and wind power prediction using hybrid empirical mode decomposition and kernel ridge regression. *Appl Soft Comput* 2018;70:1167–88.
- [11] Poggi P, Muselli M, Nottou G, Cristofari C, Louche A. Forecasting and simulating wind speed in corsica by using an autoregressive model. *Energy Convers Manag* 2003;44(20):3177–96.
- [12] Soman SS, Zareipour H, Maliki O, Mandel P. A review of wind power and wind speed forecasting methods with different time horizons. In: North American power symposium 2010. IEEE; 2010. p. 1–8.
- [13] Erdem E, Shi J. Arma based approaches for forecasting the tuple of wind speed and direction. *Appl Energy* 2011;88(4):1405–14.
- [14] Liu H, Tian H-q, Li Y-f. An emd-recursive arima method to predict wind speed for railway strong wind warning system. *J Wind Eng Ind Aerod* 2015;141:27–38.
- [15] Cao L, Qiao D, Chen X. Laplace à, “1 huber based cubature kalman filter for attitude estimation of small satellite. *Acta Astronaut* 2018;148:48–56.
- [16] Bludszuweit H, Domínguez-Navarro JA, Llombart A. Statistical analysis of wind power forecast error. *IEEE Trans Power Syst* 2008;23(3):983–91.
- [17] Shamshad A, Bawadi M, Hussin WW, Majid T, Sanusi S. First and second order Markov chain models for synthetic generation of wind speed time series. *Energy* 2005;30(5):693–708.
- [18] Peng C, Ma S, Xie X. Observer-based non-pdc control for networked t-s fuzzy systems with an event-triggered communication. *IEEE Transactions on Cybernetics* 2017;47(8):2279–87.
- [19] Cong Q, Yu W. Integrated soft sensor with wavelet neural network and adaptive weighted fusion for water quality estimation in wastewater treatment process. *Measurement* 2018;124:436–46.
- [20] Deng Y, Wang B, Lu Z. A hybrid model based on data preprocessing strategy and error correction system for wind speed forecasting. *Energy Convers Manag* 2020;212:112779.
- [21] Panja R, Pal NR. Ms-svm: minimally spanned support vector machine. *Appl Soft Comput* 2018;64:356–65.
- [22] Chang G, Lu H, Chang Y, Lee Y. An improved neural network-based approach for short-term wind speed and power forecast. *Renew Energy* 2017;105:301–11.
- [23] Wang L, Li X, Bai Y. Short-term wind speed prediction using an extreme learning machine model with error correction. *Energy Convers Manag* 2018;162:239–50.
- [24] Noorollahi Y, Jokar MA, Kalhor A. Using artificial neural networks for temporal and spatial wind speed forecasting in Iran. *Energy Convers Manag* 2016;115:17–25.
- [25] Demolli H, Dokuz AS, Ercemis A, Gokcek M. Wind power forecasting based on daily wind speed data using machine learning algorithms. *Energy Convers Manag* 2019;198:111823.
- [26] Yang Z, Long K, You P, Chow M-Y. Joint scheduling of large-scale appliances and batteries via distributed mixed optimization. *IEEE Trans Power Syst* 2014;30(4):2031–40.
- [27] Santamaría-Bonfil G, Reyes-Ballesteros A, Gershenson C. Wind speed forecasting for wind farms: a method based on support vector regression. *Renew Energy* 2016;85:790–809.
- [28] Wang H-z, Li G-q, Wang G-b, Peng J-c, Jiang H, Liu Y-t. Deep learning based ensemble approach for probabilistic wind power forecasting. *Appl Energy* 2017;188:56–70.
- [29] Zhang Z, Qin H, Liu Y, Yao L, Yu X, Lu J, Jiang Z, Feng Z. Wind speed forecasting based on quantile regression minimal gated memory network and kernel density estimation. *Energy Convers Manag* 2019;196:1395–409.
- [30] Zheng H, Yuan J, Chen L. Short-term load forecasting using emd-lstm neural networks with a xgboost algorithm for feature importance evaluation. *Energies* 2017;10(8):1168.
- [31] López E, Valle C, Allende H, Gil E, Madsen H. Wind power forecasting based on echo state networks and long short-term memory. *Energies* 2018;11(3):526.
- [32] Yan J, Zhang H, Liu Y, Han S, Li L, Lu Z. Forecasting the high penetration of wind power on multiple scales using multi-to-multi mapping. *IEEE Trans Power Syst* 2018;33(3):3276–84.
- [33] Jiao R, Huang X, Ma X, Han L, Tian W. A model combining stacked auto encoder and back propagation algorithm for short-term wind power forecasting. *Ieee Access* 2018;6:17851–8.
- [34] Liu H, Mi X-w, Li Y-f. 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 Manag* 2018;156:498–514.
- [35] Gonzalez J, Yu W. Non-linear system modeling using lstm neural networks. *IFAC-PapersOnLine* 2018;51(13):485–9.
- [36] Xiu L, Liu H, Zhang X, Kong K, Y. Lee, Wind speed forecasting using deep neural network with feature selection, *Neurocomputing*.
- [37] Begam KM, Deepa S. Optimized nonlinear neural network architectural models for multistep wind speed forecasting. *Comput Electr Eng* 2019;78:32–49.
- [38] Li C, Xiao Z, Xia X, Zou W, Zhang C. A hybrid model based on synchronous optimisation for multi-step short-term wind speed forecasting. *Appl Energy* 2018;215:131–44.
- [39] Zhang C, Wei H, Zhao J, Liu T, Zhu T, Zhang K. Short-term wind speed forecasting using empirical mode decomposition and feature selection. *Renew Energy* 2016;96:727–37.
- [40] Santhosh M, Venkaih C, Kumar DV. Ensemble empirical mode decomposition based adaptive wavelet neural network method for wind speed prediction. *Energy Convers Manag* 2018;168:482–93.
- [41] Wu Z, Huang NE. Ensemble empirical mode decomposition: a noise-assisted data analysis method. *Adv Adapt Data Anal* 2009;1(1):1–41. 0.
- [42] Yang Z, Wang J. A hybrid forecasting approach applied in wind speed forecasting based on a data processing strategy and an optimized artificial intelligence algorithm. *Energy* 2018;160:87–100.
- [43] Colominas MA, Schlotthauer G, Torres ME. Improved complete ensemble emd: a suitable tool for biomedical signal processing. *Biomed Signal Process Contr* 2014;14:19–29.
- [44] Huang NE, Shen Z, Long SR, Wu MC, Shih HH, Zheng Q, Yen N-C, Tung CC, Liu HH. The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society of London. Series A: mathematical, physical and engineering sciences* 1971;454:903–95. 1998.
- [45] Chaabi L, Lemzadmi A, Djebala A, Bouhalais ML, Ouelaa N. Fault diagnosis of rolling bearings in non-stationary running conditions using improved ceemdan and multivariate denoising based on wavelet and principal component analyses. *Int J Adv Manuf Technol* 2020;1–15.
- [46] Zhai Y, Yang X, Peng Y, Wang X, Bai K. Multiscale entropy feature extraction method of running power equipment sound. *Entropy* 2020;22(6):685.
- [47] Zhang W, Qu Z, Zhang K, Mao W, Ma Y, Fan X. A combined model based on ceemdan and modified flower pollination algorithm for wind speed forecasting. *Energy Convers Manag* 2017;136:439–51.
- [48] Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput* 1997;9(8):1735–80.
- [49] Yan K, Wang X, Du Y, Jin N, Huang H, Zhou H. Multi-step short-term power consumption forecasting with a hybrid deep learning strategy. *Energies* 2018;11(11):3089.
- [50] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using rnn encoder-decoder for statistical machine translation, arXiv preprint arXiv:1406.1078.
- [51] Wang W-c, Chau K-w, Xu D-m, Chen X-Y. Improving forecasting accuracy of annual runoff time series using arima based on eemd decomposition. *Water Resour Manag* 2015;29(8):2655–75.
- [52] Y. Sakamoto, M. Ishiguro, G. Kitagawa, Akaike information criterion statistics, Dordrecht, The Netherlands: D. Reidel 81.
- [53] Li Y, Wu H, Liu H. Multi-step wind speed forecasting using ewt decomposition, lstm principal computing, relm subordinate computing and iewt reconstruction. *Energy Convers Manag* 2018;167:203–19.
- [54] Chen H-Y, et al. Tensorflow—a system for large-scale machine learning. *OSDI* 2016;16:265–83.