

Article

# Wind Power Ramp Event Forecasting Based on Feature Extraction and Deep Learning

Li Han, Yan Qiao, Mengjie Li and Liping Shi \*

School of Electrical and Power Engineering, China University of Mining and Technology, Xuzhou 221116, China; hanli@cumt.edu.cn (L.H.); TS19130191P31@cumt.edu.cn (Y.Q.); TS20130122P31@cumt.edu.cn (M.L.)

\* Correspondence: shiliping@cumt.edu.cn; Tel.: +86-1381-328-8206

Received: 31 October 2020; Accepted: 2 December 2020; Published: 6 December 2020



**Abstract:** In order to improve the accuracy of wind power ramp forecasting and reduce the threat of ramps to the safe operation of power systems, a wind power ramp event forecast model based on feature extraction and deep learning is proposed in this work. Firstly, the Optimized Swinging Door Algorithm (OpSDA) is introduced to detect wind power ramp events, and the extraction results of ramp features, such as the ramp rate, are obtained. Then, a ramp forecast model based on a deep learning network is established. The historical wind power and its ramp features are used as the input of the forecast model, thereby strengthening the model's learning for ramp features and preventing ramp features from being submerged in the complex wind power signal. A Convolutional Neural Network (CNN) is adopted to extract features from model inputs to obtain the coupling relationship between wind power and ramp features, and Long Short-Term Memory (LSTM) is utilized to learn the time-series relationship of the data. The forecast wind power is used as the output of the model, based on which the ramp forecast result is obtained after the ramp detection. Finally, the wind power data from the Elia website is used to verify the forecast performance of the proposed method for wind power ramp events.

**Keywords:** ramp forecasting; ramp features; CNN; LSTM; Optimized Swinging Door Algorithm

## 1. Introduction

As renewable clean energy, wind energy is widely used in China. According to the statistical results released by the National Energy Administration, the installed capacity of national wind power integration has increased by 25.74 million kW in 2019, with an installed gross capacity of 210 million kW [1]. With the scale integration of wind power increasing, the safe and stable operation of the power system has been seriously affected by uncertainty. In particular, problems caused by wind power ramps, such as the imbalance of the power grid between power generation and power supply, induce great safety hazards or serious economic losses to the operation of the power grid [2]. Therefore, for the safe operation and economic dispatch of power systems, it is extremely important to be able to obtain accurate forecast results of ramp events. The difference between ramp forecasting and wind power forecasting is that the event is used as the object of ramp forecasting. There are two methods for forecasting ramps: direct forecasting and indirect forecasting.

Direct forecasting is used to obtain the detection mechanism by training historical ramp data, and then the features of ramp events (such as ramp amplitude, ramp duration, or ramp rate) are directly obtained without the need for wind power forecasting [3]. In [4], a model based on the autocorrelation statistical characteristics of ramps and a day-ahead forecast algorithm for the sequence of ramps were established. In [5], the Support Vector Machine (SVM) was used to classify and forecast the ramp amplitude directly based on historical wind power data. However, there are obvious shortcomings in

direct forecasting; that is, a large amount of historical wind power data is required to train the model, and the completeness of the ramp data in these historical data affects the forecast accuracy.

Indirect forecasting is the current mainstream method for wind power ramp event forecasting, which is based on the use of wind power forecasting to obtain ramp forecast results by ramp detection [6]. For example, the Back Propagation (BP) neural network and Radial Basis Function (RBF) neural network were used in [7,8], respectively, to forecast the components of wind power decomposed by the Atomic Sparse Method (ASD). However, because a ramp event is both a strong mutation and a low-probability event, it is difficult for traditional neural networks to learn the features of wind power when a ramp event occurs. Thus, it is difficult to obtain a high forecast accuracy. Currently, deep learning networks with stronger learning capabilities are widely used in wind power forecasting. The forecast accuracy can be improved by extracting wind power features with a Convolutional Neural Network (CNN). In [9,10], a CNN was used to extract the hidden features of the coupling relationship between wind energy, wind speed and wind direction. In [11], a CNN was used to extract spatial features from the spatial wind speed matrix. The time-series feature of wind power can be effectively learned by Long Short-Term Memory (LSTM) with a special structure to obtain more accurate forecast results [12]. In [13,14], LSTM was also used as a nonlinear mechanism for forecasting. Although the performance of wind power forecasting based on the deep learning network is improved, for ramp forecasting, the wind power ramp features are not sufficiently considered in the forecast model input, which also affects the accuracy of the ramp forecast model.

In order to solve these problems, many scholars have carried out research on the input of the forecast model and used the relevant feature data of wind power as the input of models to improve the forecast accuracy. For example, in [15], the effective meteorological variables in Numerical Weather Prediction (NWP) were utilized as the model inputs. In [16,17], the long-term trend features of wind power were captured by physical models before forecasting. In [18], not only were the spatial features of wind power considered, but the combination of a deep neural network and multi-task learning was also adopted, and the input from multiple wind farms was simultaneously received to forecast ramps based on the spatial correlation of wind farms. However, it is still uncertain whether these feature data, used as the inputs in existing studies, are directly related to ramp features.

In [19], the abrupt feature in the wind power time-series extracted by a CNN was applied as the ramp feature, and this feature was utilized as the input of LSTM to capture the long-term wind power ramp feature. However, the extraction process of ramp features based on CNN was not strictly defined. As a small sample event in a wind power sequence, ramp events are easily submerged by wind power, and the ramp feature information is quite complicated. If ramp features are not extracted according to the definition of the ramp or are not used as the input, a complete feature situation is difficult for models to learn, and thus it is hard to obtain effective results from wind power ramp event forecasting.

In summary, a wind power ramp event forecast model based on the Optimized Swinging Door Algorithm (OpSDA), a Convolutional Neural Network and Long Short-Term Memory is proposed in this paper. Firstly, OpSDA is used to detect historical wind power ramp events and extract feature values. Then, the hidden ramp features are automatically extracted by a deep learning network, which can effectively explore the coupling relationship between power and ramps. The second extraction for ramp features with CNN is conducive to the effective learning of small-sample ramp events by the forecast model. Then wind power and ramp features are used as the input of the LSTM model. Finally, the wind power forecast data are detected again to obtain the forecast results of ramp events.

## 2. Feature Extraction Based on Ramp Events

### 2.1. Features of Wind Power Ramp Events

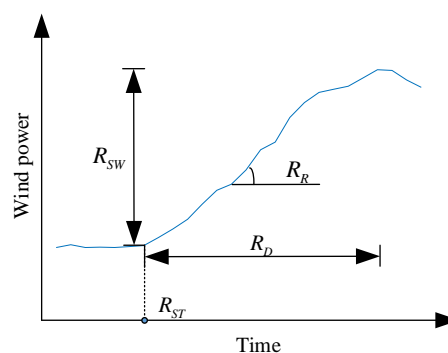
A wind power ramp event refers to a phenomenon in which wind power changes greatly in one direction in a short time. In accordance with the work presented in [2], a ramp event is defined by

Equation (1) in this paper. It indicates that a ramp event occurs when the absolute value of the difference between the power at the beginning and the end of a period of time exceeds the decision threshold.

$$|P_{t+\Delta t} - P_t| > P_{\text{threshold}}, \quad (1)$$

where  $P_{t+\Delta t}$  is the power at time  $t + \Delta t$ ,  $P_t$  is the power at time  $t$ , and  $P_{\text{threshold}}$  is the decision threshold, which is set to 3% of the installed capacity in this paper in accordance with the work in [13]. Ramps can be divided into up-ramps and down-ramps depending on the direction. An up-ramp is a sudden increase in wind power over a period of time ( $P_{t+\Delta t} - P_t > 0$ ), and a down-ramp is a sudden decrease in wind power over a period of time ( $P_{t+\Delta t} - P_t < 0$ ).

Based on the performance analysis of the wind power ramp event, four important features that define the ramp event are shown in Figure 1, which includes the ramp rate  $R_R$ , the ramp amplitude  $R_{SW}$ , the ramp start time  $R_{ST}$  and the ramp duration  $R_D$  [20].



**Figure 1.** Features of the wind power ramp event.

Wind power ramp events pose a serious threat to the power quality and safe operation of the power grid. They even cause problems such as frequency instability, load shedding and blackouts. In order to reduce the hazards caused by wind power ramps, high-precision ramp forecasting is urgently needed to provide data support for subsequent grid dispatching [21]. Due to the strong randomness and great fluctuation of wind power ramps and the stronger nonlinearity of ramps than of non-ramps, the features of the ramp segments in wind power are quite different from those of the non-ramp segments. Therefore, it is more difficult to forecast the wind power in the ramp segment, and large errors are formed by using the traditional wind power forecast method when a ramp occurs.

A forecast method that considers the influence of ramp features in the forecast process to improve the accuracy is proposed in [22]. The reasons for considering ramp features are as follows: (1) a wind power ramp event is a low-probability event for the entire wind power time-series, so the impact of ramp features on the forecast needs to be fully considered when using the deterministic model to forecast wind power; (2) four key ramp features are designed to measure whether a ramp occurs and the severity of a ramp. Therefore, a larger ramp rate when a ramp occurs means wind power with stronger nonlinearity, and forecast errors of ramps are higher than those of non-ramps. On the one hand, the ramp features are used as the input of the forecast model to learn the relationship between wind power and ramp events during the training process and to improve the forecast accuracy. On the other hand, this is also conducive to the safe, stable and economic operation of the wind power grid, which can reduce the reserve margin during grid dispatching and the impact of ramps on the power generation and power supply balance of the grid.

## 2.2. Ramp Detection and Feature Extraction Based on OpSDA

The Swinging Door Algorithm (SDA) is a data compression algorithm that filters samples according to the parallelogram rule. The parallelogram rule refers to the construction of a parallelogram based

on adjustable parameters  $\varepsilon$  (door width) to divide a long sample into multiple small samples [23]. The initial point and the end point of each segment of data are detected as SDA piecewise points, and the data in the middle of each segment are compressed. As shown in Figure 2, the first sample point A is acquired as the initial point, and multiple parallelograms are constructed in the order of samples, with  $2\varepsilon$  as the height to enclose samples, such as in the parallelogram AC and the parallelogram AD. Each time we construct a parallelogram, we need to determine whether the parallelogram covers all the samples in the segment. When the parallelogram cannot completely cover all the data in the segment, such as the parallelogram AF, the previous sample point E is acquired as the end point of the segment and the initial point of the next segment [24]. By analogy, points A, E, I and K in Figure 2 are SDA piecewise points. Points B, C and D are compressed in the first segment, points F, G and H are compressed in the second segment, and point J is compressed in the third segment.

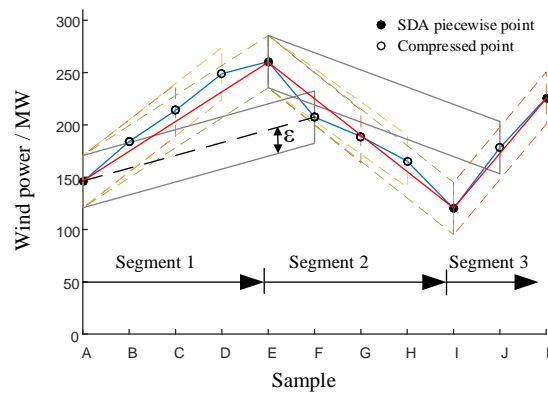


Figure 2. The basic principle of the Swinging Door Algorithm (SDA).

In SDA, only ramps are detected between the piecewise points according to Equation (1). Generally, a ramp event on the long-term scale can be divided into several adjacent small events by the SDA, and defects are present in the detection effect. Given an appropriate scoring function  $S$ , the Optimal Swinging Door Algorithm (OpSDA) is suitable for ramp detection on a long-term scale. The objective function  $P(i, j)$  is constructed for any wind power time-series in the time interval  $(i, j)$ , and ramps are recognized by obtaining their maximum. Depending on SDA piecewise points, adjacent events with the same ramp direction are classified as the same event by the optimized algorithm.

$$P(i, j) = \max(S(i, k) + P(k, j)), \forall k \in (i, j), \tag{2}$$

The constraint of the objective function  $P(i, j)$  is

$$S(i, j) > S(i, k) + S(k + 1, j), \tag{3}$$

$$S(i, j) = (j - i)^2 R(i, j), \tag{4}$$

where  $S(i, k)$  is the score of the time interval  $(i, k)$  and  $R(\cdot)$  is the ramp rule. If Equation (1) is satisfied within the time interval  $(i, j)$ , a ramp event occurs, and  $R(i, j) = 1$  is obtained; otherwise,  $R(i, j) = 0$  is obtained.

In order to improve the accuracy of ramp detection, some events, called bumps, in the wind power time-series that have a small amplitude but are opposite to the adjacent ramp direction also need to be recognized by OpSDA. Up-ramp detection is taken as an example by using Equation (5) to determine whether a bump occurs.

$$(p_{k+1} - p_k) \times [1 - B(k, k + 1)] \geq 0, \tag{5}$$

where  $p_{k+1}$  and  $p_k$  are the wind power data at time  $k + 1$  and  $k$ , respectively, and  $B(\cdot)$  is the bump rule. If a bump occurs in the time interval  $(k, k + 1)$ , then  $B(k, k + 1) = 1$ ; otherwise,  $B(k, k + 1) = 0$ . Because only up-ramps are recognized, the algorithm continues to execute at  $p_{k+1} - p_k > 0$  or stops at  $p_{k+1} - p_k < 0$ .

Assuming that the number of ramps in the sample is  $M(\forall m : 1 \leq m < M)$ , the total time interval set of ramps is  $\xi = \{t_1, \dots, t_m, \dots, t_M\}$ , where the time interval of the  $m$ -th ramp event is  $t_m = (t_m^s, t_m^e)$ .  $t_m^s$  and  $t_m^e$  are the start time and end time, respectively. The total time interval set of non-ramps is  $\bar{\xi} = \{\bar{t}_1, \dots, \bar{t}_m, \dots, \bar{t}_M\}$ ; similarly,  $\bar{t}_m = (\bar{t}_m^s, \bar{t}_m^e)$ . The wind power sequence can be expressed as the optimal  $(\Theta = \{t_1, \bar{t}_1, \dots, t_m, \bar{t}_m, t_{m+1}, \bar{t}_{m+1}, \dots, t_M, \bar{t}_M\})$  by the combination of ramps and non-ramps. In order to detect bumps, suppose that the total time interval set of bumps in the sample is  $\beta = \{\bar{t}_{1,b}, \dots, \bar{t}_{n,b}, \dots, \bar{t}_{N,b}\}$ , and the  $n$ -th non-ramp is  $\bar{t}_{n,b}$ . The wind power sequence can be signified as  $\{t_1, \bar{t}_1, \dots, t_n, \bar{t}_{n,b}, t_{n+1}, \dots, t_M, \bar{t}_M\}$ . A bump  $(\bar{t}_{n,b})$  and two adjacent ramps  $(t_n, t_{n+1})$  are integrated into one ramp event—that is,  $\bar{t}_{n,b} = (t_n^e + 1, t_n^s - 1)$ —and the number of ramps in the sample is changed into  $M - N$ .

The wind power ramp event is detected by the OpSDA, and the process of extracting its features is shown in Figure 3. For indirect forecasting, the accuracy of ramp forecasting is determined by the degree of ramp detection. Wind power data from a Belgian wind farm from 5:30 on 4 April to 2:00 on 6 April 2019—taken from the Elia website [25]—represent an example detected by OpSDA, and the results are shown in Figure 4. In the first and third up-ramps, the detected bumps and the adjacent ramps are classified as one event, which improves the shortcomings of SDA not being able to detect the long-term ramps. Therefore, the ramp trend of the wind power sequence can be effectively detected by OpSDA, which provides a data basis for the second feature extraction.

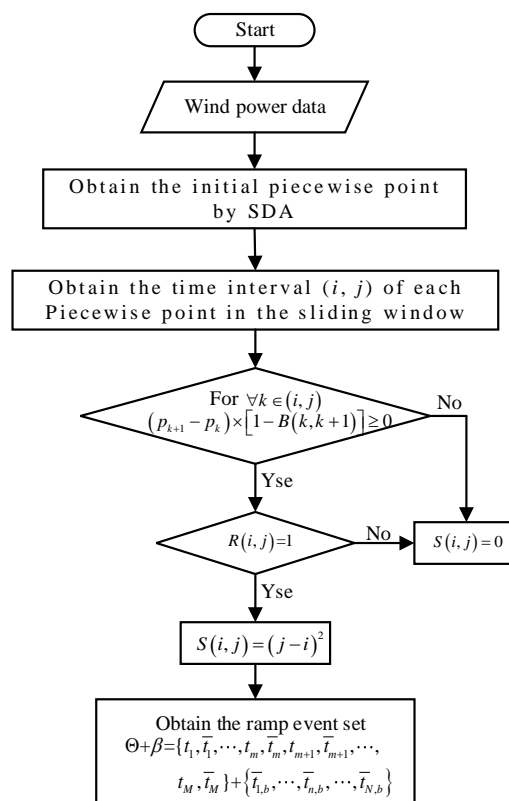


Figure 3. Flowchart of the Optimized Swinging Door Algorithm (OpSDA).

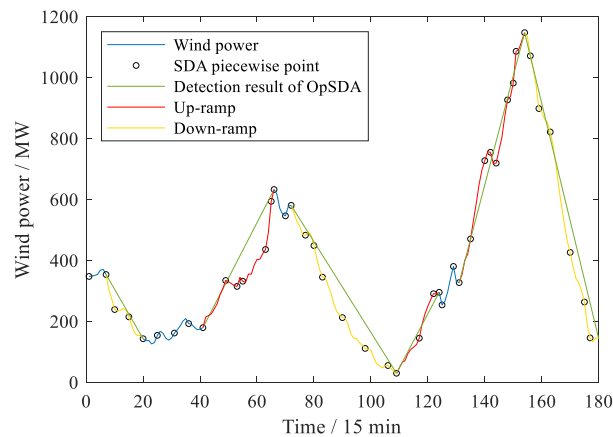


Figure 4. Ramp detection result based on OpSDA.

### 3. Ramp Forecasting Based on Deep Learning

Compared to traditional machine learning methods, a deep learning network captures data features using multiple hidden layers and emphasizes the depth of the model, which means that the hidden features of data can be mined and the forecast accuracy is effectively improved. However, it is difficult to learn complex wind power ramp features based only on one kind of certain deep learning network [26]. Therefore, the Convolutional Neural Network–Long Short-Term Memory network forecast model is proposed in this paper to deeply learn the ramp features of wind power and then obtain the forecast results of ramp events.

#### 3.1. Basic Principles of Deep Learning Network

##### 3.1.1. CNN

The CNN is a feedforward neural network and one of the representative algorithms of deep learning, with four characteristics: a local connection, weight sharing, pooling operation and a multi-layer structure [27]. Moreover, compared with a traditional neural network, the CNN can automatically extract the effective local features of the data with lower computational complexity and stronger generalization ability. In order to extract the coupling relationship between wind power and other influencing factors and the time-series relationship of features, the CNN has been widely used in the field of wind power in recent years [28].

The typical structure of a CNN usually consists of the convolutional layer, the pooling layer and the fully connected layer. In the convolution layer, the input is extracted by a convolution operation, which is expressed by Equation (6):

$$L_j^l = f\left(\sum_{i \in N_j} x_i^{l-1} * k_{ij}^l + b_j^l\right), \quad (6)$$

where the output of the  $i$ -th feature map in the  $(l-1)$ -th layer is represented by  $x_i^{l-1}$ ;  $*$  is the convolution operation;  $k_{ij}^l$  is the convolution kernel weight matrix, which is used to connect the  $j$ -th feature map in the  $l$ -th layer and the  $i$ -th feature map in the  $(l-1)$ -th layer;  $b_j^l$  is the bias matrix;  $N_j$  is the set of input feature maps;  $L_j^l$  is the output of the  $j$ -th feature map in the  $l$ -th layer; and  $f(\cdot)$  is the activation function.

The output of the convolutional layer is subjected to second feature extraction and information filtering in the pooling layer, thereby retaining the most significant features. The numbers of parameters



and data are compressed in the pooling layer, which can effectively reduce overfitting and minimize the complexity of the network. The operation in the pooling layer is shown in Equation (7):

$$L_j^l = \text{down}(x_i^{l-1}), \quad (7)$$

where  $\text{down}(\cdot)$  implicates a subsampled function.

Common pooling methods mainly include the mean-pooling with the average feature value and the max-pooling with the maximum feature value. In wind power ramp forecasting based on ramp features, for the law information of the power and ramp feature sequence to be effectively learned by the subsequent LSTM model, the max-pooling is selected to extract the most important information of ramp features.

Taking the time-series data of wind power and its four ramp features as the input, the one-dimensional convolution model is used in this paper to extract features, as shown in Figure 5.

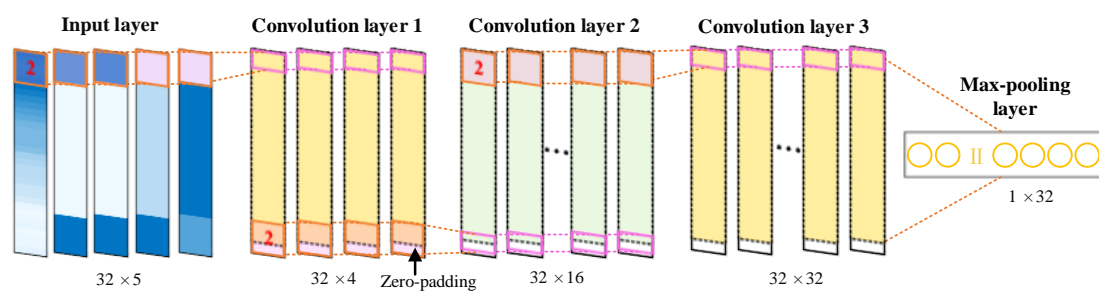


Figure 5. The structure of the Convolutional Neural Network (CNN).

According to Figure 5, the input of the model is five one-dimensional data feature maps with image height  $n$ , where  $n$  is also the sliding window width of the model. Multiple hidden layers are used for training data by the deep learning network to effectively learn essential features. In this paper, the CNN model is set with three convolutional layers and a max-pooling layer to deeply learn the data set and extract features. Features are extracted by moving the convolution kernels (filters) on feature maps for convolution operation. The size of feature maps is determined by the size of convolution kernels, which is the depth of the next layer of the CNN. The size of the convolution kernels in the first convolutional layer is  $2 \times 4$ , and the stride is 1, so the original output size should be  $(n - 1) \times 4$ . However, in order to prevent the loss of ramp feature information and keep the CNN output feature size unchanged from the size of the original feature map, all-zero padding is adopted by the model to maintain the feature dimension, setting the padding as “equal”. Thus, the output of the first convolutional layer should be  $n \times 4$ . By analogy, the size of the convolution kernel in the second convolutional layer is  $2 \times 16$ , and the output size is  $n \times 16$ . The size of the convolution kernel in the third convolutional layer is  $2 \times 32$ , and the output size is  $n \times 32$ . As the number of feature maps in each layer increases, multi-layer convolution can extract more complex features from low-level features. In the deep learning network, a nonlinear activation function  $f(\cdot)$  is also needed to construct a sparse matrix to remove the redundancy of the data and retain the features of the data as far as possible. Therefore, the model in this paper includes a rectified linear unit (ReLU) with good generalization performance after each convolutional layer. The pooling kernel size in the max-pooling layer is 32. After the feature map obtained by the convolutional layer is processed by the max-pooling layer, the final output vector size is  $1 \times n$  and forms the input of the LSTM model.

### 3.1.2. LSTM

In recent years, LSTM has been used a great deal in the field of wind power forecasting. Dynamic changes of time-series can be captured by LSTM, which can solve the problem of the long-term dependence on wind power and prevent the gradient of wind power from disappearing in the network transmission. The basic unit structure of LSTM is shown in Figure 6, including three control gates: the input gate, the forget gate and the output gate. The important ramp features can be retained by the unique memory cell of LSTM, and the unimportant features can be discarded by the forget gate to strengthen the network's learning ability for ramp features.

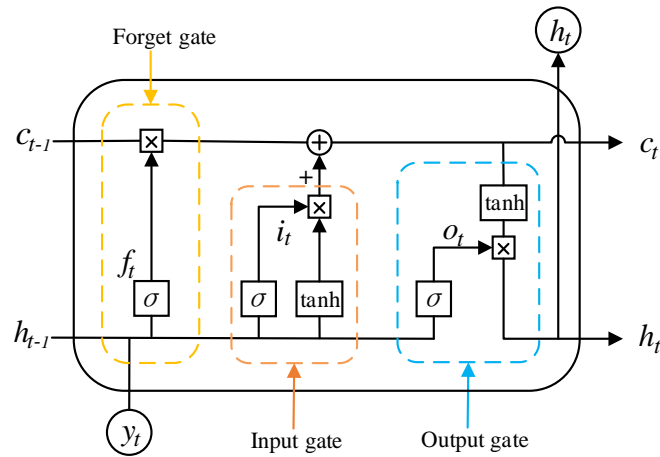


Figure 6. Basic structure of Long Short-Term Memory (LSTM).

The activation function formula [29] of each gate is

$$\begin{cases} i_t = \sigma(W_{xi}y_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \\ f_t = \sigma(W_{xf}y_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \\ o_t = \sigma(W_{xo}y_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \\ c_t = f_t c_{t-1} + i_t \tanh(W_{xc}y_t + W_{hc}h_{t-1} + b_c) \\ h_t = o_t \tanh(c_t) \end{cases}, \quad (8)$$

where  $\sigma$  is the sigmoid activation function or the tanh activation function;  $y_t$  is the input vector at time  $t$ ;  $W_{xi}$ ,  $W_{hi}$ ,  $W_{ci}$ ,  $W_{xf}$ ,  $W_{xo}$ ,  $W_{ho}$ ,  $W_{co}$ ,  $W_{xc}$  and  $W_{hc}$  are all weight parameter matrices;  $b_i$ ,  $b_f$ ,  $b_o$  and  $b_c$  are all bias vectors;  $c_t$  is the instant state vector of the state unit;  $h_t$  is the output of the state unit;  $i_t$  is the output of the input gate;  $f_t$  is the output of the forget gate; and  $o_t$  is the output of the output gate.

### 3.2. Structure Design of CNN-LSTM

On the basis of Section 3.1, if only wind power is utilized as the input, it is difficult for the forecast model to capture the ramp features of wind power. Therefore, historical wind power and ramp features corresponding to the time-series extracted by OpSDA are utilized as the input of the CNN-LSTM deep learning network model established in this paper. They are constructed as a feature map by the sliding time window, and the wind power to be forecasted is used as the output. Thus, the learning of the relationship between wind power and ramp features is deepened in the network to reduce the ramp forecast error. In the deep learning network model, a CNN is adopted for the second feature extraction for wind power and ramp features, and LSTM is adopted to forecast wind power. The structure of the CNN-LSTM model is shown in Figure 7.



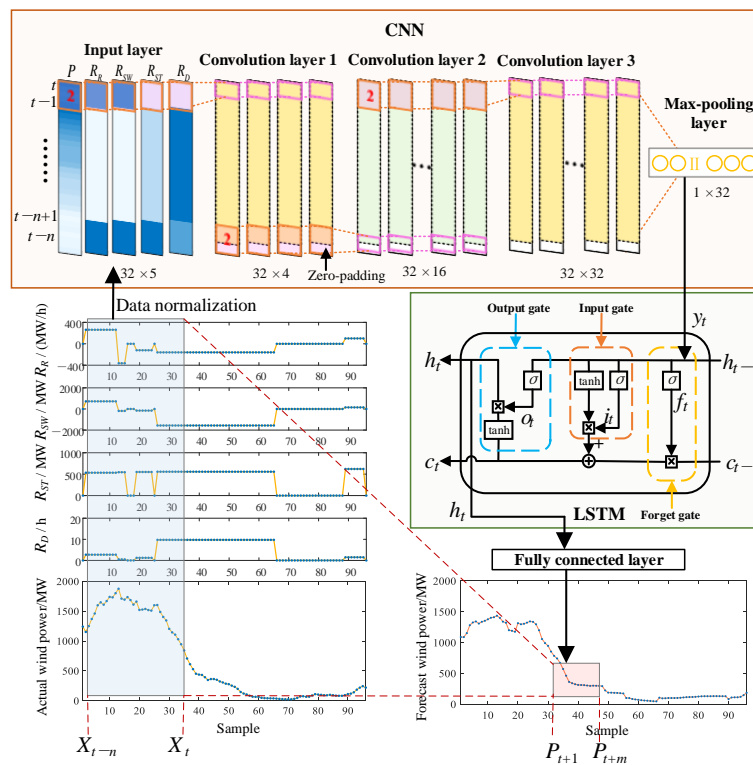


Figure 7. Structure of CNN-LSTM.

(1) The input

Wind power and four ramp features form a data set that is used as the input. The four ramp features are the ramp rate  $R_R$ , the ramp amplitude  $R_{SW}$ , the ramp start time  $R_{ST}$  and the ramp duration  $R_D$ . In order to maintain the continuity of the ramp features, the features are filled with zeros in non-ramps according to the original time-series sequence, meaning that the feature dimension is consistent with the wind power dimension. In Figure 7,  $P$  is the wind power data from the Belgian Elia website, with a data point every 15 min.  $t$  is any moment in the wind power,  $m$  is the forecast ahead step and  $n$  is set to 32, which is the width of the sliding window. Thus, the input data set is  $\{X_t, X_{t-1}, \dots, X_{t-n}\}$ , where  $X_t = \{P_t, (R_R)_t, (R_{SW})_t, (R_{ST})_t, (R_D)_t\}$ .

(2) Feature extraction based on CNN

As shown in Figure 7, three convolutional layers and a max-pooling layer are used in the CNN model in this paper for deep learning and feature extraction. Since a wind power ramp event is a contingency, for continuous wind power sequence data, discontinuity is a feature of the ramp feature data. The convolution kernels of CNN operate on the original feature map and perform convolution operations at the same time so that the potential relationship of each data point in the feature map can be effectively extracted to form feature vectors. Furthermore, the CNN model can start from a small area and extract the ramp features in the entire dataset layer-by-layer from the deep structure by deep learning.

A batch of wind power samples taken from the Elia website in January 2020. Figure 8a–c is drawn from the three convolutional layers based on the samples. The horizontal axis represents the samples arranged in the order of sampling time, and the sample time interval is 15 min; the vertical axis represents the normalized wind power value at that moment, and the light pink background indicates that a ramp event occurred in this segment.

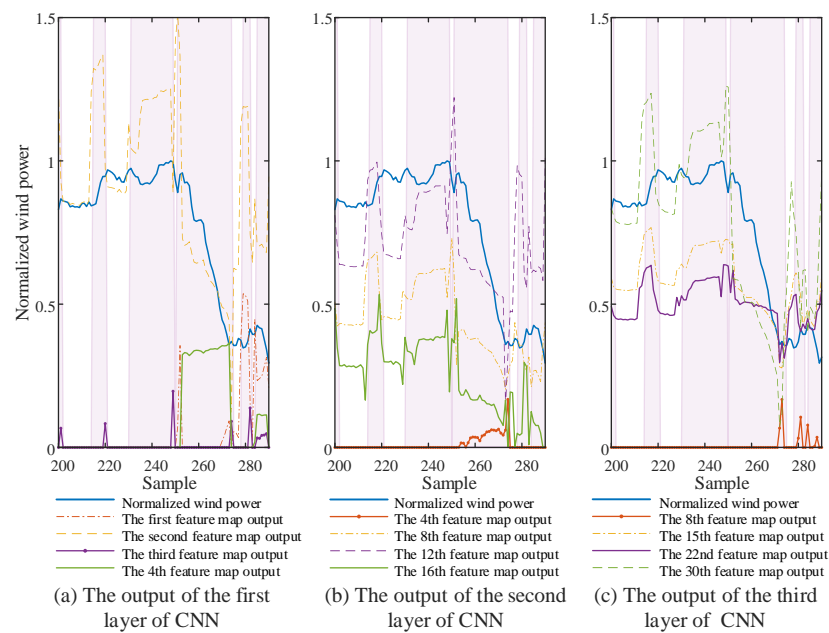


Figure 8. Output of convolutional layers.

According to Figure 8, there are mutations in the feature map output of the three convolutional layers near the samples where the ramp occurs. From Figure 8a—the first convolutional layer—due to the small number of convolution kernels, the more accurate ramp feature is only extracted by the second feature map output. The output of the remaining three feature maps basically shows no ramp trend. In Figure 8b, features are extracted by the 16 convolution kernels in the second convolution layer, and the ramp trend can basically be described by the feature map outputs. However, the fluctuation range of the outputs is large and the concentration is weak. According to Figure 8c, after features are extracted by the third convolutional layer, the ramp trend of the outputs is more concentrated. The outputs of the 15th, 22nd and 30th feature maps show a corresponding mutation when a ramp event occurred. Among them, the 30th feature map output includes a strong mutation, while the features basically stabilize when there is no ramp. Therefore, it can be seen that there is a certain relationship among adjacent samples of wind power ramp features. The ability of CNN to capture local trends is conducive to learning the relationship and extracting wind power ramp features.

### (3) Forecasting based on LSTM

Due to the time-series relationship of samples that can be learned by LSTM, ramp features extracted by the CNN model and wind power are used as the input of the LSTM model so that the relationship between ramps and the wind power can be determined. To learn the data rules without over-fitting, a single-layer LSTM network is used, where the number of neurons is set to be 128, and the wind power forecast results are exported by the fully connected layer.

In order to meet the dispatching requirements, it is necessary to forecast the long-term wind power ramp situation in advance. As shown in Figure 7,  $X_{t-n}, X_{t-n+1}, \dots, X_t$  are the data input to the model each time and  $P_{t+1}, P_{t+2}, \dots, P_{t+m}$  are the outputs. According to National Energy Administration documents [30], wind farms report the forecast curve for the next 4 h every 15 min. In other words, to forecast 16 future wind power data points, the forecast ahead step is usually set to be  $m = 16$ . To further improve the forecast accuracy, this paper adopts the rolling multi-step forecast method, and the forecast data are updated every 15 min. Only the first forecast point is retained for 4 h-ahead forecasting.

The structure of the CNN–LSTM model established in this paper is shown in Table 1, and end-to-end training is applied to the model.

**Table 1.** The structure of the CNN–LSTM model.

Layer (Type)	Output Shape	Param
input_1 (Input Layer)	(None, 32, 5)	0
conv1d (Conv1D)	(None, 32, 4)	44
conv1d_1 (Conv1D)	(None, 32, 16)	144
conv1d_2 (Conv1D)	(None, 32, 32)	1056
max_pooling1d (MaxPooling1D)	(None, 1, 32)	0
lstm (LSTM)	(None, 128)	82,432
dense (Dense)	(None, 16)	2064

In summary, the overall idea of multi-step forecasting based on ramp features and deep learning proposed in this paper is as follows:

Step 1: OpSDA is used to detect historical wind power, and four types of ramp features are extracted;

Step 2: A sliding window is used to divide the input, composed of wind power and ramp features, that is input into the CNN–LSTM model. The ramp features are extracted by the CNN again, and 16 steps-ahead forecast results are forecast by the LSTM. The final rolling multi-step wind power forecast result is obtained by extracting the first forecast point of the multi-step forecasting;

Step 3: Based on the ramp detection of the forecast power, the forecast results of ramp events are obtained.

## 4. Case Study

### 4.1. Ramp Forecasting Evaluation Indexes

According to [2], there are four situations for ramp forecast results: forecasting a ramp to occur that does not actually occur; forecasting a ramp not to occur that actually occurs; forecasting a ramp to occur that actually occurs; and forecasting a ramp not to occur that does not actually occur. The occurrence times of the four situations are represented by  $N_{FP}$ ,  $N_{FN}$ ,  $N_{TP}$  and  $N_{TN}$  in turn. There are many evaluation indexes for ramp forecasting, and they have not been uniformly specified. Therefore, this paper selects multiple indexes to evaluate the performance of the forecast model from different perspectives.

The recall is defined by

$$R_C = \frac{N_{TP}}{N_{TP} + N_{FN}}, \quad (9)$$

the precision is defined by

$$P_C = \frac{N_{TP}}{N_{TP} + N_{FP}}, \quad (10)$$

the bias index is defined by

$$B_S = \frac{N_{TP} + N_{FP}}{N_{TP} + N_{FN}}, \quad (11)$$

the critical success index is defined by

$$C_{SI} = \frac{N_{TP}}{N_{TP} + N_{FN} + N_{FP}}, \quad (12)$$

the accuracy is defined by

$$A_{CC} = \frac{N_{TP} + N_{TN}}{N_{TP} + N_{FN} + N_{FP} + N_{TN}}, \quad (13)$$

the missing report rate is defined by

$$M_I = \frac{N_{FN}}{N_{TP} + N_{FN} + N_{FP} + N_{TN}}, \quad (14)$$

the false alarm rate is defined by

$$E_R = \frac{N_{FP}}{N_{TP} + N_{FN} + N_{FP} + N_{TN}}, \quad (15)$$

the up-ramp forecast accuracy rate is defined by

$$S_R = \frac{C_R}{N_R}, \quad (16)$$

and the down-ramp forecast accuracy rate is defined by

$$S_{NR} = \frac{C_{NR}}{N_{NR}}, \quad (17)$$

where  $C_R$  and  $C_{NR}$  are, respectively, the number of correct forecast up-ramps and down-ramps;  $N_R$  and  $N_{NR}$  are, respectively, the total number of actual up-ramps and down-ramps.

The mean absolute percentage error is defined by

$$I_{MAPE} = \frac{1}{n} \sum_{t=1}^n \frac{|P_{actual} - P_{forecast}|}{\bar{P}_{actual}}, \quad (18)$$

where  $n$  is the number of wind power samples,  $P_{actual}$  is the actual wind power value,  $P_{forecast}$  is the forecast wind power value and  $\bar{P}_{actual}$  is the average of the actual wind power.

The recall indicates the probability of the forecast occurrence of the ramp in terms of its actual occurrence. The precision represents the probability that the ramp is forecasted to occur and that it then occurs in reality. The bias index expresses the ratio of the correct forecast number of the ramp to the actual number. The critical success index measures the accuracy of ramp forecasting and indicates the validity of the forecast results. The accuracy is the probability of correctly forecasting the wind power event, which reflects the accuracy of the model in the forecasting of ramps. The missing report rate is the probability of ramp actually occurring that is forecasted not to occur. The false alarm rate is the probability of forecasting a non-ramp as a ramp. The accuracy rate of up-ramp and down-ramp forecasting evaluates the performance of the model from different ramp directions. The mean absolute percentage error is a common index used to evaluate the accuracy of the deterministic wind power forecast model. This paper is about ramp forecasting, and thus in order to better evaluate the model, the mean absolute percentage error is only analyzed when a ramp occurs.

#### 4.2. Ramp Detection and Feature Extraction

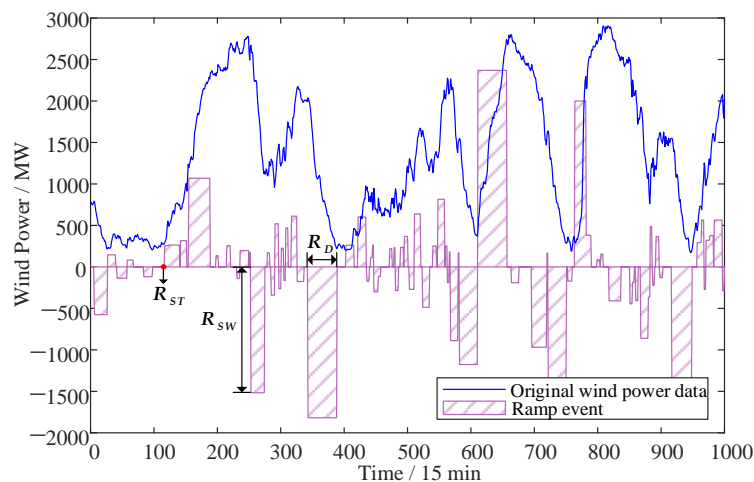
In this section, the wind power data from the Elia Belgian wind farm with an installed capacity of 3796 MW in January 2020 are used to detect ramps. The ramp detection and feature extraction based on OpSDA were realized in MATLAB R2019b. The detection results are shown in Table 2. The start time in Table 2 is expressed in a numeric format by using “datenum”. Due to space limitations, only the features of the first 24 ramp events in January are displayed.

The 1000 wind power samples from the Elia Belgian wind farm in January 2020 are taken as an example, and the results of the ramp features detected and extracted by OpSDA are shown in Figure 9. Each rectangle in Figure 9 represents a ramp event. The rectangle in the first quadrant represents an up-ramp, and that in the fourth quadrant represents a down-ramp. The abscissa in Figure 9 represents the duration of each ramp and the sample point of the wind power at the beginning of the ramp. The ordinate represents the amplitude of each ramp.

**Table 2.** Ramp events and their features.

Ramp	Direction	$R_R$ /(MW/h)	$R_{SW}$ /MW	$R_{ST}$	$R_D$ /h
1	↓	115.05	575.24	737,791.0521	5.00
2	↑	45.11	146.60	737,791.2604	3.25
3	↓	36.46	136.72	737,791.4271	3.75
4	↑	33.18	82.96	737,791.5833	2.50
5	↓	36.42	118.37	737,791.8646	3.25
6	↑	205.04	256.30	737,822.2083	1.25
7	↑	62.10	263.91	737,822.2708	4.25
8	↑	140.85	316.91	737,822.4688	2.25
9	↑	129.48	1068.22	737,822.6042	8.25
10	↑	180.97	135.73	737,851.0521	0.75
11	↑	204.59	255.74	737,851.2292	1.25
12	↓	140.65	140.65	737,851.3958	1.00
13	↑	56.32	197.12	737,851.4375	3.50
14	↑	353.36	176.68	737,851.6042	0.50
15	↓	288.88	1516.6	737,851.6250	5.25
16	↑	227.20	170.4	737,851.8958	0.75
17	↓	272.34	340.43	737,851.9583	1.25
18	↑	297.13	519.98	737,882.0104	1.75
19	↓	351.00	263.25	737,882.0833	0.75
20	↑	180.30	225.38	737,882.1146	1.25
21	↓	328.36	164.18	737,882.1667	0.5
22	↑	469.53	469.53	737,882.1875	1.00
23	↑	271.94	611.87	737,882.2917	2.25
24	↓	70.296	175.74	737,882.3854	2.5

‘↑’ means an up-ramp and ‘↓’ means a down-ramp.

**Figure 9.** The effect graph of ramp features detected and extracted by OpSDA.

#### 4.3. Performance Analysis of Ramp Forecast Model

Python was adopted as the programming language by the CNN–LSTM forecast model in this paper. The compiling environment was PyCharm Community Edition 2020, 16 GB of RAM was used, and the processor was an AMD Ryzen 7 4800H. The forecast model implementation of this paper was the Keras deep learning framework, which can be used as a high-level application program interface for TensorFlow, Theano and Microsoft Cognitive Toolkit (Microsoft-CNTK), enabling rapid model construction and experimental development. The forecast performance of CNN–LSTM was analyzed by using the data from the Elia website. The first 500 points and the second 500 points of the data of each quarter from October 2019 to September 2020 were used to train and test the network, respectively.

Additionally, in this section, the model performance is analyzed with different parameters and with other models to verify the feasibility.

### 4.3.1. Ramp Forecast Performance with Different Parameters

According to the process of ramp detection, the wind power data are firstly compressed by OpSDA based on the  $\epsilon$ -value, and multiple SDA piecewise points are obtained. Ramp detection and subsequent ramp forecasting are affected to a certain extent by the  $\epsilon$ -value. In the multi-step forecast model, different forecast results are attained due to different forecast-ahead steps.

In conclusion, to evaluate the different forecast results obtained with different parameters and to show the effect of considering the ramp features in the input of the forecast model to improve the accuracy, data from January 2020 are used for analysis. The performance of the multi-step ramp forecast model with different forecast-ahead steps, different door widths and different inputs is evaluated by four indexes:  $R_C$ ,  $P_C$ ,  $B_S$  and  $C_{SI}$ . Based on Equations (19) and (20), the relationship between the four evaluation indexes is shown in Figure 10.

$$C_{SI} = \frac{1}{\frac{1}{R_C} + \frac{1}{P_C} - 1}, \tag{19}$$

$$B_S = \frac{R_C}{P_C}, \tag{20}$$

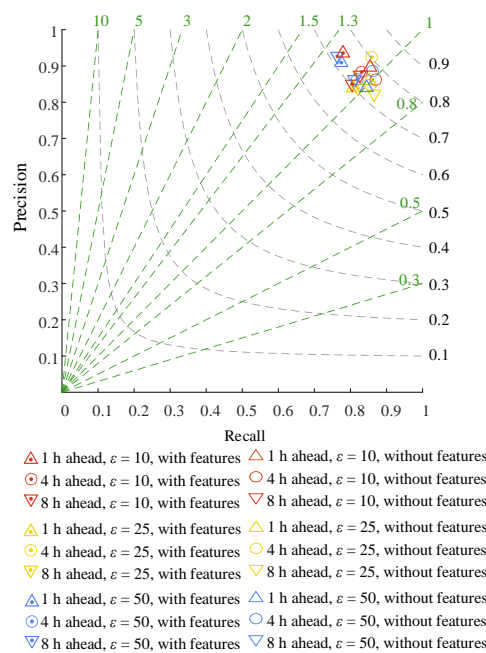


Figure 10. Forecast model performance with different parameters.

The forecast effect of the model can be displayed intuitively according to the evaluation index results in Figure 10. The abscissa represents the precision  $P_C$ , the ordinate represents the recall  $R_C$ , the gray dashed line represents the critical success index  $C_{SI}$  and the green dashed line represents the bias index  $B_S$ . Moreover, the better the forecast effect, the closer the evaluation index is to the upper right corner. The recall results of the model input including the ramp features are generally higher than these of only power, and  $R_C$  reaches more than 0.9 many times. When  $\epsilon = 25$ , the evaluation points are mainly distributed near the diagonal line, which shows that the model can simultaneously obtain higher recall and higher precision than  $\epsilon = 50$  and  $\epsilon = 10$ . The evaluation results of 16 forecast-ahead steps (4 h) are arranged more closely; thus, the forecast effect is more stable. Combined with the

statistical results in [31], about 95% of wind power ramp events last less than 4.04 h, so the forecast ahead step is set to 16 steps (4 h) in this paper.

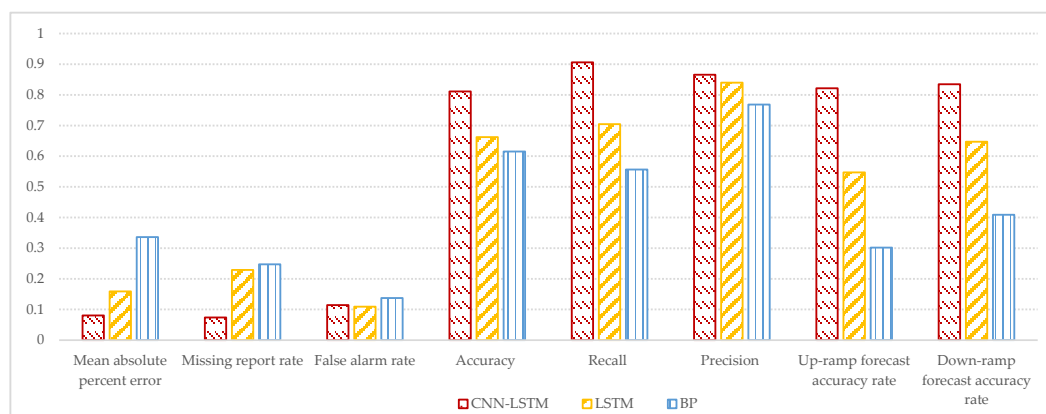
The yellow solid point in Figure 10 is the optimal parameter point. The input of this point contains ramp features, a forecast-ahead step that is set to 16 steps (4 h) and a door width that is set to be  $\varepsilon = 25$ . In this case, the precision is 0.8587. In addition, not only is the probability of correct results, accounting for all ramp forecast results, higher, but the probability of the correct forecasting of ramp occurrence also reaches the optimal result for  $P_C$ , and the  $R_C$ -result is 0.9240. Furthermore, a higher  $B_S$  and better  $C_{SI}$  are obtained in the meanwhile. In summary, the case with the optimal parameters is adopted in this paper for the CNN-LSTM forecast model. In actual engineering applications, different optimal parameters can be set for forecasting according to different wind power data.

#### 4.3.2. Performance Analysis of Different Forecast Models

In this section, BP and LSTM are compared with the proposed model in this paper by using data from the ELIA website to evaluate the performance of wind power ramp forecasting. The wind power data are detected by OpSDA and its ramp features are extracted, and they are used as the input together for the CNN-LSTM model, the LSTM model and the BP model for training. In the data from October 2019 to September 2020, 1000 data points are taken every quarter. The first 500 are used as the training set and the last 500 are used as the test set. To analyze the performance of the forecast model,  $I_{MAPE}$ ,  $ACC$ ,  $M_I$ ,  $E_R$ ,  $R_C$ ,  $P_C$ ,  $S_{NR}$  and  $S_R$  are obtained and shown in Table 3. Based on Table 3, the annual average of each model for each index is obtained and shown in Figure 11 so that the forecast performance of the three models can be more intuitively displayed.

**Table 3.** Comparison of evaluation indexes with different forecast models in each quarter. BP: Back Propagation.

Model	Quarter	$I_{MAPE}$	$ACC$	$M_I$	$E_R$	$R_C$	$P_C$	$S_{NR}$	$S_R$
CNN-LSTM	Q1	0.0748	0.8274	0.0575	0.1150	0.9240	0.8587	0.8769	0.8480
	Q2	0.0880	0.8407	0.0819	0.0774	0.9051	0.9098	0.8172	0.9020
	Q3	0.0603	0.8451	0.0973	0.0575	0.8725	0.9205	0.7941	0.7847
	Q4	0.0982	0.7323	0.0597	0.2080	0.9221	0.7729	0.7974	0.8041
LSTM	Q1	0.1274	0.7588	0.1482	0.0929	0.8040	0.8675	0.7464	0.7892
	Q2	0.1463	0.7611	0.1239	0.1150	0.8564	0.8652	0.6559	0.77451
	Q3	0.1155	0.5465	0.4049	0.0487	0.4695	0.8804	0.4853	0.4163
	Q4	0.2464	0.5819	0.2389	0.1792	0.6888	0.7468	0.3006	0.6082
BP	Q1	0.3152	0.6055	0.2260	0.1684	0.6997	0.7576	0.3043	0.3721
	Q2	0.4012	0.5756	0.3624	0.0618	0.575	0.8880	0.2737	0.4143
	Q3	0.3837	0.6268	0.2196	0.1535	0.7065	0.7750	0.2721	0.4279
	Q4	0.2448	0.6524	0.1812	0.1663	0.2448	0.6524	0.3562	0.4221



**Figure 11.** Comparison of annual average evaluation indexes with different forecast models.



As shown in Table 3, the  $I_{MAPE}$  results of the CNN–LSTM model for four quarters at the ramp are all less than 0.1. The smaller the mean absolute percentage error, the better the wind power forecasting effects. Here, the wind power forecast effect in the third quarter is the best and the  $I_{MAPE}$  result is only 0.0603. Besides, the  $I_{MAPE}$  results of the LSTM model and the BP model are both significantly larger than the proposed ramp forecast model. The other seven indexes mainly reflect the accuracy of the forecast model for ramp forecasting. In the first three quarters, the  $A_{CC}$  results of the CNN–LSTM forecast model are above 0.82. Although the  $A_{CC}$  result of CNN–LSTM in the fourth quarter is slightly insufficient, it is still higher than that of the other two forecast models. Based on  $S_R$  in Table 3 reaching 0.8041, the model is proven to be useful for up-ramp forecasting. There are cases in which the missing report rate  $M_I$  or the false alarm rate  $E_R$  of the BP model and the LSTM model is lower than that of the CNN–LSTM model, as in the third quarter, but the  $A_{CC}$  result of these two methods is far inferior to the CNN–LSTM model. In the CNN–LSTM model, higher accuracy can be obtained when the missing report rate and false alarm rate are relatively low. Furthermore, in the fourth quarter, higher  $R_C$  and  $P_C$  values are obtained by the CNN–LSTM model, which are both above 0.9 in the second quarter; that is, the number of correctly forecasted ramps accounts for more than 90% of the total number of forecasted ramps and the total number of actual ramps. In general, the forecast accuracy of each model for down-ramp forecasting is higher than that for up-ramp forecasting, and up-ramps are more difficult to forecast than down-ramps.

According to the definition of each evaluation index, the requirements for the evaluation index of a model with good ramp forecast performance are as follows: a high accuracy  $A_{CC}$ , recall  $R_C$ , precision  $P_C$ , up-ramp forecast accuracy rate  $S_{NR}$  and down-ramp forecast accuracy rate  $S_R$ ; and a low mean absolute percentage error  $I_{MAPE}$ , missing report rate  $M_I$  and false alarm rate  $E_R$ . As shown in Figure 11, the annual average  $P_C$  values of BP and LSTM are both above 0.75, and the annual average  $E_R$  value is below 0.14. However, according to the annual average results of other indexes, the above index requirements are not satisfied; for example, due to the low  $S_{NR}$ -result. For the CNN–LSTM model, the results of the three indexes on the left of Figure 11 are all below 0.12, and the results of the five indicators on the right in Figure 11 are all above 0.81. Moreover, the annual average  $R_C$  is as high as 0.9059. Therefore, ramp forecast results that are more stable and satisfy the above index requirements can be obtained based on the CNN–LSTM model.

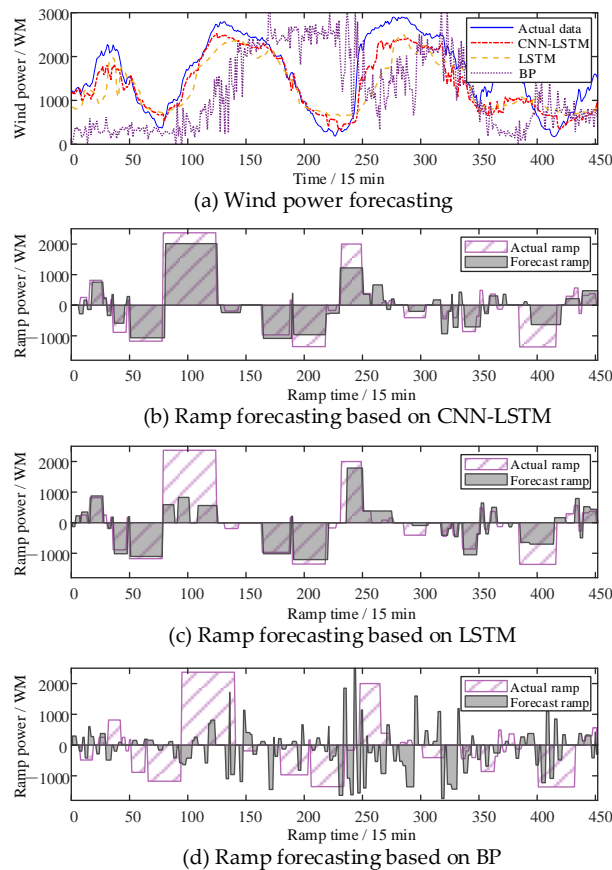
In order to further verify the performance of the wind power ramp forecast model, based on the same evaluation indexes, the optimal results forecasted by the model in this paper are compared with those of three recent ramp forecast research works, as shown in Table 4. The three results are based on the multilayer perception–boosted trees (MLP–BT) model, the multilayer perception–Markov switching auto-regression (MLP–MSAR) model, and the deep neural networks–Multi-Task Learning model with four shared layers and a specification layer (DNN–MTL(C4S1)). Although the overall forecast accuracy  $A_{CC}$  of the proposed ramp forecast model is slightly lower than that of the MLP–BT model and the MLP–MSAR model, the proposed ramp forecast model can balance the various evaluation indexes better and can obtain a better precision  $P_C$  while ensuring a higher recall  $R_C$ , and each index result is higher than 0.81. In addition,  $R_C$ ,  $P_C$  and  $S_R$  are all above 0.9, so the performance of the proposed ramp forecast model is better for forecasting up-ramps and down-ramps.

**Table 4.** Comparison of evaluation indexes with models in other research works.

Model	$A_{CC}$	$R_C$	$P_C$	$S_{NR}$	$S_R$
CNN–LSTM	0.8407	0.9051	0.9098	0.8172	0.9020
MLP–BT [16]	0.8669	0.9286	0.8125	—	—
MLP–MSAR [17]	0.8600	0.8889	0.8571	—	—
DNN–MTL(C4S1) [18]	0.7133	—	—	0.6567	0.7047

The ramp forecast results in January 2020 based on CNN–LSTM are shown in Figure 12. Figure 12a shows the 16-step-ahead wind power forecast results obtained from the original wind power and ramp

features based on three models. Based on these results, the final ramp forecast results are obtained after being re-detected by OpSDA, as shown in Figure 12b–d. According to Figure 12, the forecast effect of the model is shown more intuitively by the display method of the ramp forecast results proposed in this paper. The ramp amplitude and the ramp time are acquired as the forecast results to provide a better dispatch basis for the power grid.



**Figure 12.** Ramp forecasting results based on CNN-LSTM.

According to Figure 12, severe fluctuations are found in the forecast results of the LSTM model and the BP model. In particular, the long-term ramp event is forecasted by the BP model as multiple events with a short ramp duration but large ramp amplitude. As shown in Figure 12b, it can be clearly concluded that the CNN-LSTM model has better performance in terms of forecasting the ramp amplitude and the ramp duration. After the ramp features are extracted for the second time by the CNN, the continuity of the wind power and the ramp can be effectively learned by LSTM. Ramp forecast results for long-term wind power can be obtained by CNN-LSTM, which is conducive to the safe operation of the power system and economic dispatch.

## 5. Conclusions

Aiming at ramp features and the long-term trend features of wind power, a wind power ramp forecast method based on feature extraction and deep learning is proposed in this paper. Firstly, the historical wind power is detected by OpSDA to obtain the historical ramp features. Then, the CNN-LSTM forecast model is established, in which historical ramp features and power are used as the input and the forecast power is used as the output. Deep learning is applied to explore the coupling relationship between ramp features and wind power. Finally, the ramp forecast results are obtained by re-detecting the forecast power.

In this paper, the ramp features can be effectively extracted by CNN from the model input, meaning that the relationship between wind power and ramp events can be learned by LSTM more effectively to obtain more accurate ramp forecast results. The forecast performance of the proposed model with different parameters is discussed and optimal parameters are obtained, which means that high precision and high recall can be obtained simultaneously. Based on the evaluation indexes of ramp forecasting, the wind power samples from the Elia website are used for case analysis. It can be concluded that the evaluation results based on the CNN–LSTM model meet the index requirements, and the annual average recall is as high as 0.9059. In the end, the comparison with other research results verifies the high precision and effectiveness of the deep learning network model proposed in this paper.

**Author Contributions:** All of the authors have contributed to this research. L.H. and L.S. provided the research ideas and guided the revision of the draft; Y.Q. and M.L. performed the case studies and analysis and wrote and revised the paper. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by [the National Natural Science Foundation of China] grant number [61703404] and [the National Natural Science Foundation of China] grant number [62076243].

**Acknowledgments:** This work was supported by the National Natural Science Foundation of China 61703404 and the National Natural Science Foundation of China 62076243.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. National Energy Administration. Grid-Connected Operation of Wind Power in 2019. Available online: [http://www.nea.gov.cn/2020-02/28/c\\_138827910.htm](http://www.nea.gov.cn/2020-02/28/c_138827910.htm) (accessed on 28 February 2020).
2. Zhang, D.; Dai, Y.; Zhang, X.; Zhang, J.; Wang, Z.; Xue, L. Review and prospect of research on wind power ramp events. *Power Syst. Technol.* **2018**, *42*, 1783–1792.
3. Zheng, H.; Kusiak, A. Prediction of wind farm power ramp rates: A data-mining approach. *J. Sol. Energy Eng.* **2009**, *131*, 376–385. [CrossRef]
4. Qu, Y.; Xu, J.; Jiang, S.; Liu, Y.; Sun, Y.; Ke, D. Statistical characteristics modeling and forecasting of ramp events based on frequent patterns mining. *Autom. Electr. Power Syst.* **2020**. Available online: <http://kns.cnki.net/kcms/detail/32.1180.TP.20200713.1547.006.html> (accessed on 15 July 2020).
5. Zareipour, H.; Huang, D.; Rosehart, W. Wind power ramp events classification and forecasting: A data mining approach. In Proceedings of the IEEE Power & Energy Society General Meeting, Detroit, MI, USA, 10 October 2011.
6. Ouyang, T.; Zha, X.; Qin, L.; Xiong, Y.; Huang, H. Wind power ramps prediction method based on amendment of similar events. *Proc. CSEE* **2017**, *37*, 572–581.
7. Cui, M.; Sun, Y.; Ke, D. Wind power ramp events forecasting based on atomic sparse decomposition and BP neural networks. *Autom. Electr. Power Syst.* **2014**, *38*, 6–11.
8. Huang, Q.; Wang, Z.; Du, B.; Sheng, G.; Liu, S. Wind power ramp forecasting based on the front decomposition combination forecast method. *Renew. Energy Reso.* **2016**, *34*, 1847–1852.
9. Yin, H.; Ou, Z.; Chen, D.; Meng, A. An ultra-short-term wind power prediction based on two-layer mode decomposition and cascaded deep learning. *Power Syst. Technol.* **2019**, *39*, 2093–2106.
10. Chen, J.; Zhu, Q.; Shi, D.; Li, Y.; Zhu, L.; Duan, X.; Liu, Y. A multi-step wind speed prediction model for multiple sites leveraging spatio-temporal correlation. *Proc. CSEE* **2019**, *39*, 2093–2106.
11. Yin, H.; Ou, Z.; Huang, S.; Meng, A. A cascaded deep learning wind power prediction approach based on a two-layer of mode decomposition. *Energy* **2019**, *189*, 116316. [CrossRef]
12. Zhu, Q.; Li, H.; Wang, Z.; Chen, J.; Wang, B. Short-term wind power forecasting based on LSTM. *Power Syst. Technol.* **2017**, *41*, 3797–3802.
13. Hu, Y.; Liang, C. A nonlinear hybrid wind speed forecasting model using LSTM network, hysteretic ELM and differential evolution algorithm. *Energy Convers. Manag.* **2018**, *173*, 123–142. [CrossRef]
14. Yuan, X.; Chen, C.; Min, J.; Yuan, Y. Prediction interval of wind power using parameter optimized Beta distribution based LSTM model. *Appl. Soft Comput.* **2019**, *82*, 105550. [CrossRef]

15. Ouyang, T.; Zha, X.; Qin, L.; Xiong, Y.; Xia, T.; Huang, H. Prediction method of wind power ramp events by using meteorological conditions to select adjacent points. *Power Syst. Technol.* **2015**, *39*, 3266–3272.
16. Ouyang, T.; Huang, H.; He, Y. Ramp events forecasting based on long-term wind power prediction and correction. *IET Renew. Power Gen.* **2019**, *13*, 2793–2801. [[CrossRef](#)]
17. Ouyang, T.; Zha, X.; Qin, L.; He, Y.; Tang, Z. Prediction of wind power ramp events based on residual correction. *Renew. Energy* **2019**, *136*, 781–792. [[CrossRef](#)]
18. Dorado-Moreno, M.; Navarin, N.; Gutiérrez, P.A.; Prieto, L.; Sperduti, A.; Salcedo-Sanz, S.; Hervás-Martínez, C. Multi-task learning for the prediction of wind power ramp events with deep neural networks. *Neural Netw.* **2020**, *123*, 401–411. [[CrossRef](#)]
19. Jing, H.; Han, L.; Gao, Z. Wind power ramp forecast based on convolutional neural network for feature extraction. *Autom. Electr. Power Syst.* **2020**. Available online: <http://kns.cnki.net/kcms/detail/32.1180.TP.20200713.1548.010.html> (accessed on 15 July 2020).
20. Sevlian, R.; Rajagopal, R. Detection and statistics of wind power ramps. *IEEE Trans. Power Syst.* **2013**, *28*, 3610–3620. [[CrossRef](#)]
21. Ma, H.; Li, C.; Liu, Y. Assessing impact of wind power ramp events on operation adequacy of power systems. *Autom. Electr. Power Syst.* **2017**, *41*, 41–47.
22. Gan, D.; Ke, D.; Sun, Y.; Cui, M. Short-term probabilistic wind power forecast considering ramp characteristics. *Electr. Power Autom. Eq.* **2016**, *36*, 145–150.
23. Ye, L.; Lu, P.; Teng, J.; Zhai, B.; Wu, L.; Lan, H.; Zhang, W. Power prediction and correction model considering wind power ramping events. *Autom. Electr. Power Syst.* **2019**, *43*, 49–58.
24. Cui, M. Study of Wind Power Ramping Events. Ph. D. Thesis, Wuhan University, Wuhan, China. Ph.D. Thesis, Wuhan University, Wuhan, China, December 2015.
25. Elia. Wind-Power Generation Data. Available online: <http://www.elia.be/en/grid-data/power-generation/wind-power> (accessed on 20 October 2020).
26. Rodger, J.A.; George, J.A. Triple bottom line accounting for optimizing natural gas sustainability: A statistical linear programming fuzzy ILOWA optimized sustainment model approach to reducing supply chain global cybersecurity vulnerability through information and communications technology. *J. Clean Prod.* **2017**, *142*, 1931–1949. [[CrossRef](#)]
27. Zhou, F.; Jin, L.; Dong, J. Review of convolutional neural network. *J. Comput. Appl.* **2016**, *36*, 2508–2515.
28. Lu, J.; Zhang, Q.; Yang, Z.; Tu, M.; Lu, J.; Peng, H. Short-term load forecasting method based on CNN-LSTM hybrid neural network model. *Autom. Electr. Power Syst.* **2019**, *43*, 131–137.
29. Han, L.; Qiao, Y.; Jing, H. Interval estimation of wind power based on error classification. *Autom. Electr. Power Syst.* **2020**. Available online: <http://kns.cnki.net/kcms/detail/32.1180.TP.20200729.1639.008.html> (accessed on 30 July 2020).
30. National Energy Administration. Interim measures for the management of wind farm power forecast and forecast. *Solar Energy* **2011**, *14*, 6–7.
31. Ouyang, T.; Zha, X.; Qin, L.; Xiong, Y.; Zhu, X. Selection of time window to predict wind power ramp events. *Power Syst. Technol.* **2015**, *39*, 414–419.

**Publisher’s Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).