

2-D regional short-term wind speed forecast based on CNN-LSTM deep learning model

Yaoran Chen^a, Yan Wang^a, Zhikun Dong^a, Jie Su^a, Zhaolong Han^{a,b,c,d,*}, Dai Zhou^{a,b,c,d,*}, Yongsheng Zhao^{a,b}, Yan Bao^{a,b,c,d}

^a School of Naval Architecture, Ocean & Civil Engineering, Shanghai Jiao Tong University, Shanghai 200240, PR China

^b State Key Laboratory of Ocean Engineering, Shanghai Jiao Tong University, Shanghai 200240, PR China

^c Key Laboratory of Hydrodynamics of Ministry of Education, Shanghai Jiao Tong University, Shanghai 200240, PR China

^d Shanghai Key Laboratory for Digital Maintenance of Buildings and Infrastructure, Shanghai Jiao Tong University, Shanghai 200240, PR China



ARTICLE INFO

Keywords:

Regional wind speed prediction
CNN
LSTM
Temporal series fitness
Spatial distribution

ABSTRACT

Short-term wind speed forecast is of great importance to wind farm regulation and its early warning. Previous studies mainly focused on the prediction at a single location but few extended the task to 2-D wind plane. In this study, a novel deep learning model was proposed for a 2-D regional wind speed forecast, using the combination of the auto-encoder of convolutional neural network (CNN) and the long short-term memory unit (LSTM). The 12-hidden-layer deep CNN was adopted to encode the high dimensional 2-D input into the embedding vector and inversely, to decode such latent representation after it was predicted by the LSTM module based on historical data. The model performance was compared with parallel models under different criteria, including MAE, RMSE and R^2 , all showing stable and considerable enhancements. For instance, the overall MAE value dropped to 0.35 m/s for the current model, which is 32.7%, 28.8% and 18.9% away from the prediction results using the persistence, basic ANN and LSTM model. Moreover, comprehensive discussions were provided from both temporal and spatial views of analysis, revealing that the current model can not only offer an accurate wind speed forecast along timeline (R^2 equals to 0.981), but also give a distinct estimation of the spatial wind speed distribution in 2-D wind farm.

1. Background and introduction

Wind energy has undergone rapid development over the past two decades [1]. Meanwhile, the high fluctuation and occasional cessation of wind power have posed large uncertain factors to its grid integration [2]. To mitigate such instability and to enhance the power production, an accurate real-time forecast of the wind speed for the wind farm is of great importance [3].

The approaches for wind speed forecasts can mainly be categorized into two types: the physical model and the data-driven model [4]. The physical model is based on numerical weather forecasting (NWP) that uses various meteorological and geological information as input and then calculates the wind speed through physical laws (e.g. N-S equation) [5,6]. Nevertheless, such method is considerably expensive in its time cost, hence it is not suitable for short-term wind prediction problem [7]. On the contrary, data-driven models are more feasible for real-time forecasting because all of the required data is the recorded wind series

in history [8,9]. Purely based on the past dataset, the wind speed forecast is conducted through different algorithms to establish their statistical correlations in this methodology [10].

So far, data-driven models for wind speed prediction have undergone three phases of development. In the first era, the traditional statistical models were used for stationary time series [11], where a typical family was the auto-regression (AR) models. Erdem et al. [12] adopted the autoregressive integrated moving average (ARIMA) models to predict the hourly mean wind attributes, and compared the prediction performance with different pre-processing techniques. Tong et al. [13] combined the wavelet transformation with AR model, and respectively forecasted the wind speed 1, 3 and 7 h in advance. Such AR models are easy-to-implement, but are weak when handling non-stable wind series since they can hardly establish the non-linear relations between variables in a direct way [14].

Aiming at this problem, various machine learning (ML) models were proposed, forming the second wave in this field [15]. Khosravi et al. [16]

* Corresponding authors at: School of Naval Architecture, Ocean & Civil Engineering, Shanghai Jiao Tong University, Shanghai 200240, PR China.

E-mail addresses: han.arkey@sjtu.edu.cn (Z. Han), zhoudai@sjtu.edu.cn (D. Zhou).

developed and compared 7 machine learning models based on different combinations of algorithms, including multi-layer perceptron (MLP), support vector machine (SVM), fuzzy network (FN) and heuristic searching methods. The feasibility of the models was tested on various wind speed series with intervals from 5 to 30 min. Liu [17] et al. adopted empirical mode decomposition (EMD) and extreme learning machine (ELM) to predict the non-stationary part of the raw wind series, and they synthesized the results with the stationary outcome forecasted by ARIMA, presenting a higher performance over any other sub-models [17]. Hence, these machine learning models can effectively build the non-linear mappings from inputs to outputs and can forecast the wind speed in a closer future when given a high-resolution dataset. Yet, the prediction accuracy of naive machine learning models would approach its upper limit when the scale of historical data is continuously increasing and its feature dimension is further expanding [18].

To overcome this setback, as the third phase, different deep learning (DL) models have been developed and widely used in recent years. Peng et al. [19] developed a DL neural network with wavelet soft threshold denoising (WSTD) and gated recurrent unit (GRU). The forecasts outperformed other benchmarks under evaluations of accuracy, compatibility and computational cost. Malik et al. [20] proposed a novel deep reinforcement learning model for this task, with its mean squared error (MSE) decreased over 30% compared with the parallel ML models. Aly [21] investigated the combinations of various basic ML models into more sophisticated DL models and found out the best hybrid on target dataset. All of these investigations have shown the great potential of deep learning algorithms to solve the wind speed forecasting problem with high precision.

However, whatever the algorithms were, the above works mainly focused on the wind speed estimated at a single location, and there were few previous researches that conducted the wind speed forecast for the whole region of a 2-D wind farm that integrates bunch of sites together. As an evident merit, such synchronized regional wind speed prediction is more beneficial for the overall power grid regulation and the resource management. In addition, during the process of such regional wind speed prediction, both spatial and temporal information of wind field and turbine sites could be fully used, and would further enhance the overall prediction performance. Therefore, it is a challenging and meaningful gap to be filled.

In this study, a novel deep learning model was proposed to solve the regional wind speed forecasting problem. To begin with, as pre-processing, the orthogonalization was used to decompose the raw wind speed of each site in the target region [12], aiming to eliminate the influence of the anisotropy of wind directions among different points on the prediction result. Next, the deep learning algorithm comprising two models were adopted: first, from the view of space-correlation, the convolutional neural network (CNN) [22,23] was used as an auto-encoder to extract the low-dimension deep features from the 2D wind speeds matrix over the wind farm; second, from the temporal aspect of view, the long short-term memory network (LSTM) [24,25] was employed to forecast such deep feature based on the historical data, which would then be transferred back into the CNN decoder to make predictions of wind speed components. Eventually, these forecasted components were reversely summed up as the final wind speed [12].

The contribution of this work can be summarized as follows:

1. Instead of conducting on a single-site wind speed prediction, a novel deep learning framework was proposed for the wind speed forecasting for a 2-D regional wind farm.
2. The spatial and temporal information of the wind farm was learnt by CNN and LSTM modules respectively, and these two algorithms are coupled together in the whole framework.
3. Multi-aspect comparisons were conducted to examine the model performance, including benchmarks of parallel models and multi-criterion assessments, hence the reliability of model is improved.

4. Abundant analytical discussions were elaborated in detail from the all-round comparisons on both time-series fitness and spatial wind speed distribution.

The roadmap of this work is constructed as follows: in Section 2, we introduce the target regional wind farm dataset; in Section 3, the methodology is presented; In Section 4, we analyze the numerical results; finally, in Section 5, solid conclusions are made.

2. Dataset

The target dataset was extracted from a three-year wind attributes in a 10×10 wind array located at Indiana, US, 2010–2012. Such wind resources were publicly available in the Wind Integration National Dataset (WIND) provided by NREL [26] and have been widely cited in previous researches under various background of usage [27–29].

As shown in Fig. 1, the target wind sites were arranged in an inclined square. The orientation of the square was mildly rotated from the East-South-West-North (ESWN) system (less than 10 degree). Here, we denoted the N-S direction as the X axis, the rows of sites are indexed by 1 to 10 from the north-most to the south-most. In addition, columns of sites from the west-most to east-most were indexed by 1 to 10 in Y axis. Such convention is throughout this work.

Besides, an overview of the statistical information of the array is provided in Table 1 and Fig. 2. As shown in Table 1, there are 13140×100 points of recorded data throughout three years in total. Although the time resolution was rather high in raw dataset (5 min) [26], we diluted the time interval to be every 2 h in this work, which could be more sufficient and reasonable for the wind grid regulation. The distance between each adjacent sites of the grid was 2 km.

Fig. 2 also illustrates the distribution of mean value, standard deviation among sites and a full-time wind history of arbitrary site in the target region. As it can be referred, the average wind speed for sites ranges from 7.19 m/s to 7.50 m/s, with a minor difference between adjacent points. Meanwhile, there is a general trend of the wind speed change, where the maximum average is presented at the east-south corner (10, 10) while the low-speed zone appears in the opposite direction. Also, similar tendency can be observed in the deviation plot, presenting a potential spatial correlation to be considered.

3. Methodology

The overall flow chart of the proposed model is shown in Fig. 3, with the necessary matrix shape of the dataset denoted within brackets. In this problem, the first 80% of the whole wind history was chosen as the training series to be fed into the CNN-LSTM model, and the rest part was the testing set to assess and polish the model. The detailed illustration of data-processing and CNN-LSTM algorithm is presented in Sections 3.1 and Section 3.2, respectively.

3.1. Pre-processing

As mentioned above, the aim of orthogonalization processing on raw wind dataset is to eliminate the inconsistency of the wind direction during the wind speed regression process [12]. Given the wind speed array $X^t = (X_{ij}^t)_{m \times n}$ and the corresponding wind direction array $D^t = (D_{ij}^t)_{m \times n}$ ($m = n = 10$), taking the positive directions of north and east, the components of ESWN decomposition is described as below:

$$\begin{cases} X_{NS}^t = X^t * \cos(D^t) \\ X_{EW}^t = X^t * \sin(D^t) \end{cases}, \quad (1)$$

where each entry of D^t is a degree of wind direction, rotating anti-clockwise from the north.

Inversely, for the post-processing, if we denote the forecasted NS and

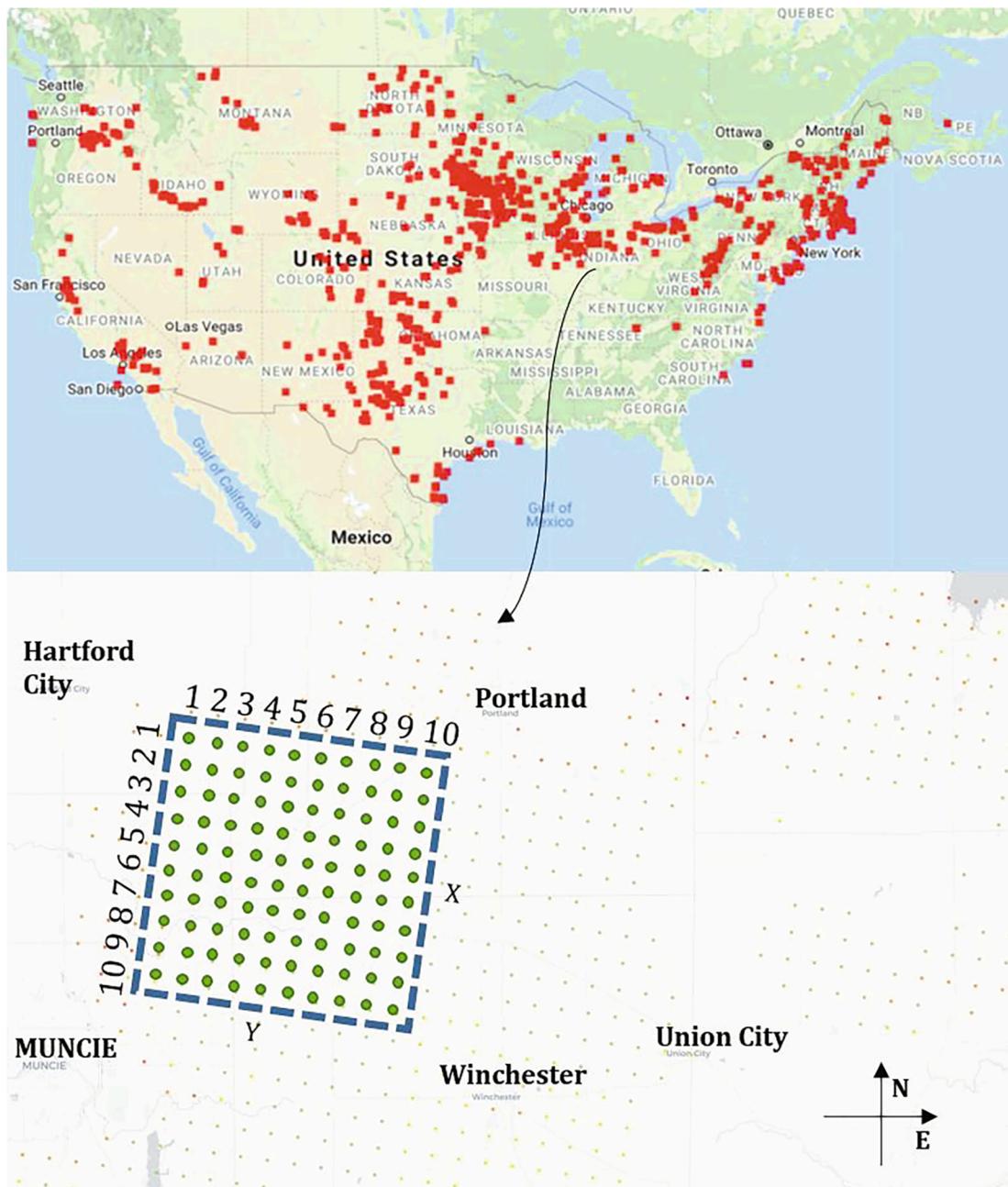


Fig. 1. Location source of regional wind speed dataset [26].

Table 1

Statistical information of the regional wind dataset.

Number of temporal points	Time step	Number of spatial points	Spatial interval	Max wind speed value	Min wind speed value
13,140	2 h	100	2 km	27.23 m/s	0.01 m/s

EW components as \widehat{X}_{NS}^{t+1} and \widehat{X}_{EW}^{t+1} , the final prediction \widehat{X}^{t+1} will be the magnitude of their vector summation:

$$\widehat{X}^{t+1} = \sqrt{\left(\widehat{X}_{NS}^{t+1}\right)^2 + \left(\widehat{X}_{EW}^{t+1}\right)^2} \quad (2)$$

3.2. CNN-LSTM algorithm

In this section, the CNN-LSTM algorithm is introduced in detail. The whole framework is illustrated in Fig. 4. As it can be found, in this system, two main modules, namely CNN and LSTM, were coupled together. The CNN worked as an extractor and translator for the deep feature. Such deep feature integrated the spatial information of the wind attributes among sites and had lower dimension compared with the raw matrix. In addition, the LSTM model acted as the temporal predictor to forecast such deep feature along the timeline under the given time step. In this work, out of the engineering consideration, the time step was chosen to be 6, meaning that the records of the past half-day were used to forecast the regional wind speed in 2 h later.

It should also be noticed that in Fig. 4, the entire dataset was divided twice. As mentioned in Section 2, to train the LSTM predictor, the raw time series was separated at a ratio of 4:1. However, when training CNN

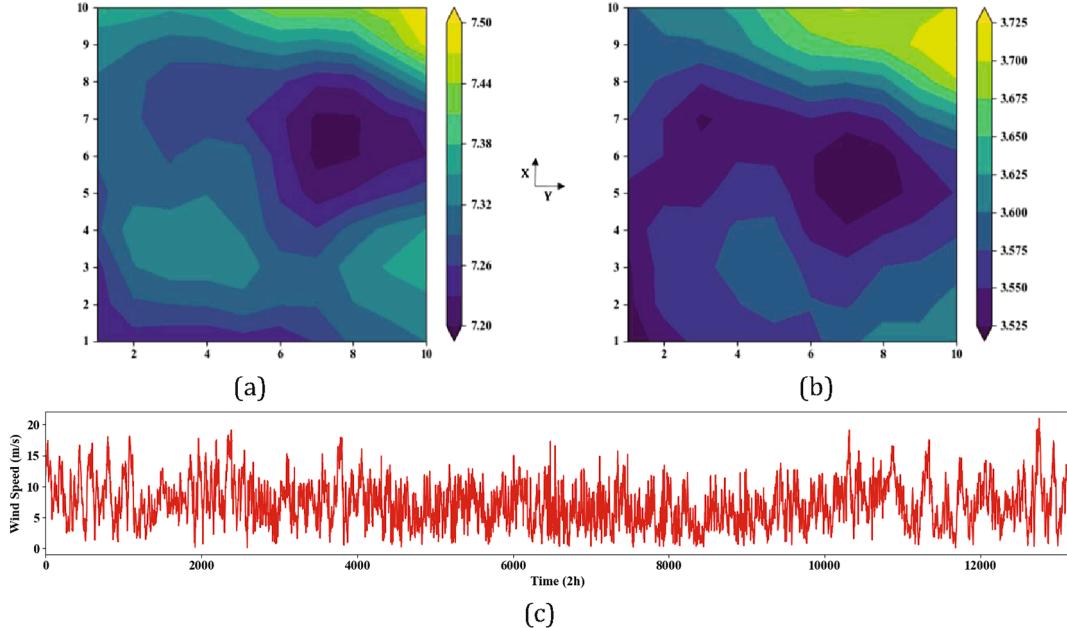


Fig. 2. Statistical overview of the regional wind speed dataset in wind farm of Indiana, USA, 2010–2012 [26]: (a). Mean speed value; (b) Standard deviation; (c) full-time wind history of a random selected site, Location: (3, 4).

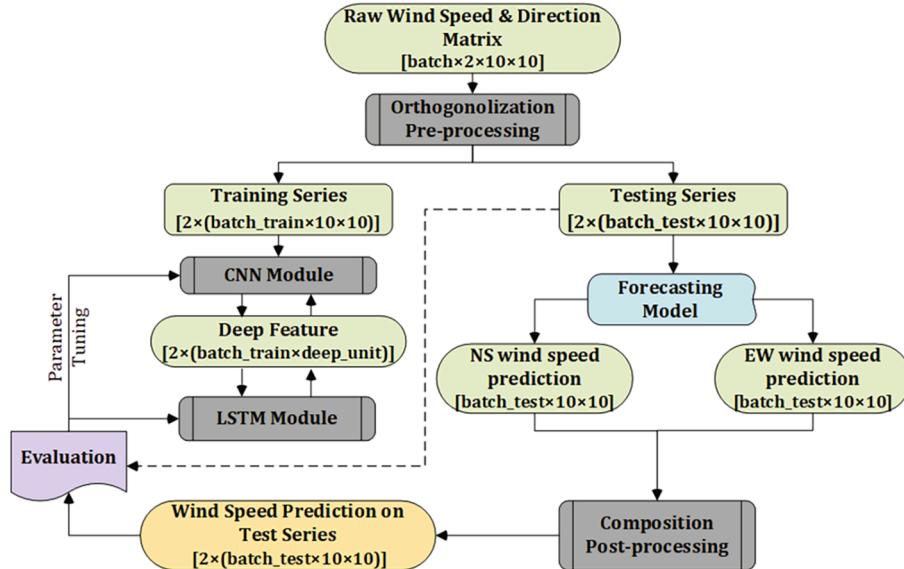


Fig. 3. Flow chart of the prediction process.

model, testing series were not used to prevent data leakage. Hence the training series were further divided into a CNN training set and CNN test set by another 4:1.

3.2.1. Convolutional neural network (CNN)

Inspired from the well-known VGG-net [30,31], the structure of CNN model is shown in Fig. 5. Small sized kernel filters with shape of 3×3 were repeatedly used, followed by the pooling layer for deep feature extraction. Working as an auto-encoder, the input and output data of CNN model were identical, and the real production was the structure itself together with the extracted 1-D deep feature containing the spatial information. The mechanism of techniques including padding, convolutional filtering and pooling are briefly introduced as follows.

Fig. 6 explains how padding and convolutional filter works [32]. Given the input matrix x with shape $D@A \times A$ and convolutional filter F_k

with shape $D@F \times F$, the output y_k is calculated as:

$$y_k = \sigma(F_k * x + b_k), k = 1, 2 \dots K \quad (3)$$

where “*” denotes the convolutional operator; b_k is the bias for the k^{th} filter; K is the total number of filters during this operation; σ is the non-linear activation function, using rectified linear unit (Relu) in this work.

Pooling is a process of down-sampling, which can effectively reduce the dimension of the matrix window, while retaining the deep information at the same time [30]. In this work, the max pooling was used. As shown in Fig. 7, given the kernel size (e.g. 2 in figure), the maximum value of each slice of each block is extracted and combined to form the output. Also, the depth of matrix does not change during the process.

3.2.2. Long Short-term memory network (LSTM)

The LSTM neural network is a robust version of recurrent neural

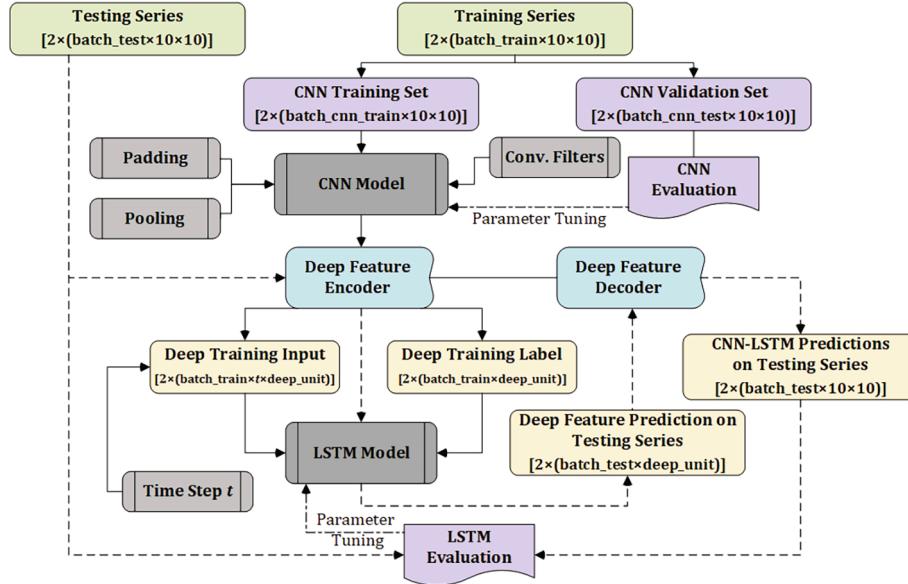


Fig. 4. Framework of CNN-LSTM model.

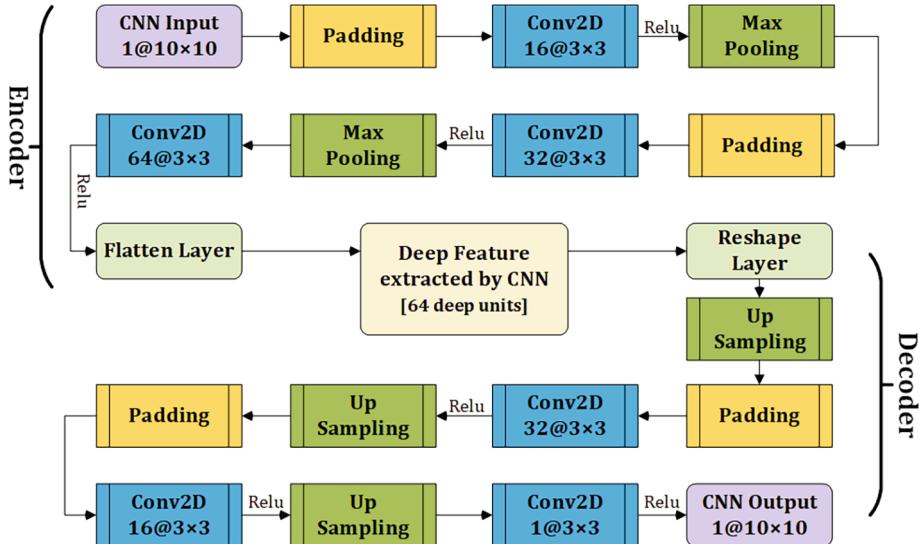


Fig. 5. Flowchart of CNN model as an auto-encoder.

network (RNN) to deal with time-series problem. The most significant contribution of LSTM is the innovation of “gates” to control its dependency on historical information, which are forget gate, input gate and output gate. Assumed that input deep feature series are $p = (p_1, p_2, \dots, p_t, \dots, p_n)$ and the output series are $q = (q_1, q_2, \dots, q_t, \dots, q_n)$, the procedure of calculation is shown in Fig. 8 as follows [33]:

$$f_t = \text{sigmoid}(\mathbf{W}_f \cdot [\mathbf{h}_{t-1}, p_t] + \mathbf{b}_f)$$

$$i_t = \text{sigmoid}(\mathbf{W}_i \cdot [\mathbf{h}_{t-1}, p_t] + \mathbf{b}_i)$$

$$C_t = \tanh(\mathbf{W}_C \cdot [\mathbf{h}_{t-1}, p_t] + \mathbf{b}_C)$$

$$o_t = \text{sigmoid}(\mathbf{W}_o \cdot [\mathbf{h}_{t-1}, p_t] + \mathbf{b}_o)$$

$$C_t = f_t^* C_{t-1} + i_t^* C_t$$

$$\mathbf{h}_t = o_t^* \text{softsign}(C_t)$$

$$q_t = \text{sigmoid}(\mathbf{W}_y \cdot \mathbf{h}_t + \mathbf{b}_y),$$

where f_t , i_t and o_t respectively represents the forget gate, input gate and output gate function. The expression of the non-linear activators (\tanh , sigmoid and softsign) can be found in Ref. [35].

3.2.3. Performance criteria

The general performance of the current model is provided in this section. To evaluate this, the following three criteria for regression task were used in this work. It should be noticed that, the first two losses (i.e. MAE and RMSE) focused on whole spatial-temporal data while the third one (i.e. R^2) was for an arbitrary site over the timeline:

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |X_i - \hat{X}_i| \quad (5)$$

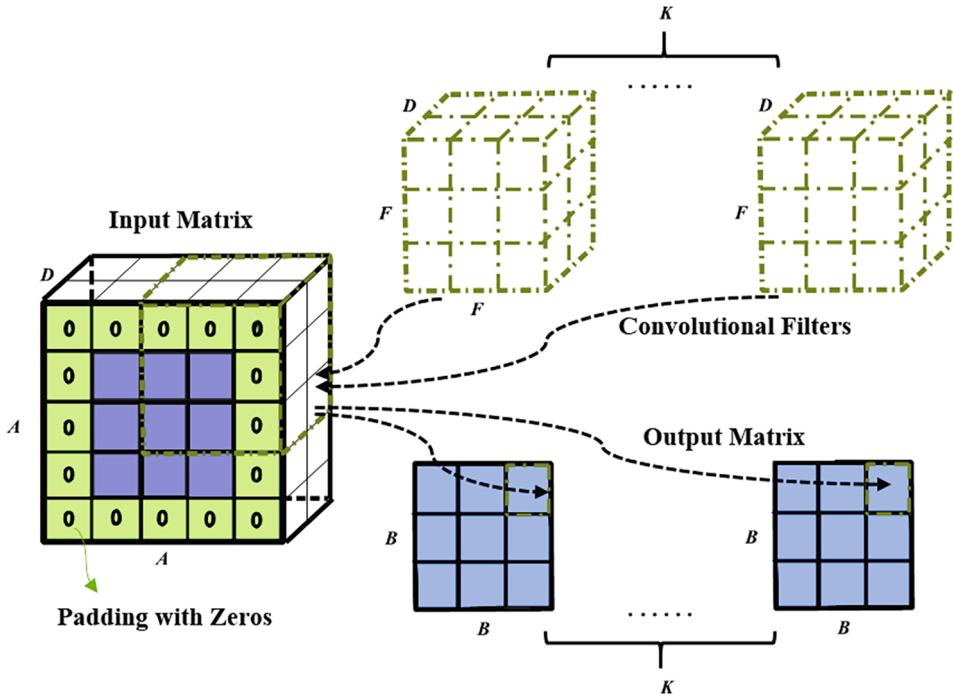


Fig. 6. Illustration of padding and convolutional computation in CNN.

21	17	0	12
5	10	7	33
14	18	30	19
16	22	32	20

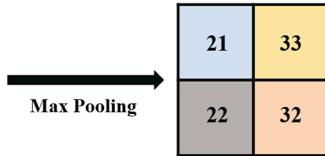


Fig. 7. Illustration of max pooling in CNN.

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (X_i - \hat{X}_i)^2} \quad (6)$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (X_i - \hat{X}_i)^2}{\sum_{i=1}^n (X_i - \bar{X}_i)^2}, \quad (7)$$

where n is the number of timestamp in test series; N is the number of wind speed data, equalling the product of the number of timestamps in test series and the number of sites; X, \hat{X} are the observed and predicted wind speed for each site over the wind history; \bar{X} is the mean value of wind speed for the site.

4. Results and discussion

In this section, the model performance is investigated through the comparison with mainstream models. In Section 4.1, the evaluations are presented in terms of different error standards. Further, in Sections 4.2

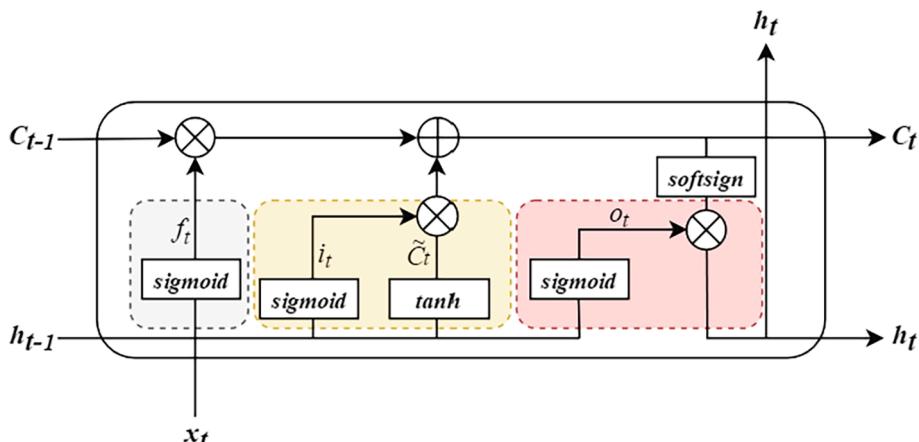


Fig. 8. Structure of LSTM unit with three gates [34].

and 4.3, the results are elaborately analyzed and discussed from temporal and spatial aspects of view.

The numerical studies of this work were performed on Python 3.8 platform with keras [34] deep learning package on the server of: (CPU: Intel(R) Core(TM) i7-7500 @2.70 GHz; GPU: 1080 Ti 11G).

4.1. General performance

Under the standards in Section 3.2.3, the comparative analysis was conducted with three benchmarks: the persistence model, the back-propagate ANN model and the LSTM model, which typically represented the model of linear, basic non-linear, non-linear with temporal information and non-linear with both temporal and spatial information, respectively. It should be noticed that, these models were carefully designed and trained. For instance, the ANN model had 3 hidden layers with totally 350 units (activation function: tanh) and adopted a small initial learning rate of $1e-4$ with the Adam [36] optimizer. Also, the LSTM model adopted two LSTM layers of 50 units (activation function: softsign) and one fully-connected layer as the output layer. The overall trainable parameters for ANN and LSTM were over 50,000. The prediction results for the persistence model, ANN, LSTM and CNN-LSTM are shown below in Fig. 9 and Table 2:

It can be found that, starting from the linear benchmark, the prediction error is continuously decreasing when the model gradually becomes sophisticated. The MAE value impressively dropped to 0.35 m/s for the current model, decreasing by 32.7%, 28.8% and 18.9% from persistence, ANN and LSTM model. In addition, under the criteria of RMSE, since this standard is sensitive to the large individual errors, the assessment value of four models all witnessed an increase. However, there was still a large decrease of the proposed model from the parallel models, accounting for 32.3%, 26.0% and 18.3%, respectively.

These results demonstrate that, for a target wind farm region, taking temporal and spatial correlation of wind speed into consideration can effectively enhance its overall prediction accuracy. To further analysis this, discussions on those two aspects are presented in the next sections.

4.2. Temporal comparison

To investigate the degree of the temporal change of prediction series fitting with the observation, the determination coefficient R^2 is used for each site.

The comparison among the mean values of R^2 for all sites in different models is shown in Fig. 10 as below. As it can be found, similar to the trend in Fig. 9, the fitness of the prediction to the observation witnessed a continuous climbing as the model evolved from linear. Out of all algorithm, the predictions from CNN-LSTM model could best explain the

Table 2
Comparison of accuracy between different models.

Models	Persistence criteria	ANN	LSTM	CNN-LSTM (Proposed model)
MAE (m/s)	0.5219	0.4932	0.4328	0.3509
RMSE (m/s)	0.7677	0.7013	0.6362	0.5193

variation of real wind history.

In addition, three sites among total 100 sites are selected for further analytical study, which are separately located at the north-east (No. (2, 9)), middle (No. (5, 6)), and south-west (No. (9, 2)) part of the wind farm array.

To clearly show the time variation, the curves are zoomed into several one-week series, respectively from the beginning, middle and ending of the testing time series for different sites, as in Fig. 11:

It can be drawn from Fig. 11, though all models could generally reproduce the trend of real wind variation, the R^2 of the current model was higher than any other model in each site and its prediction curve (blue line) was closer to the observation (red line). In Fig. 11(a), there was a sharp drop followed by an abrupt increase of wind speed in the latter half of the series, such changes were accurately predicted by current model while other curves were right-shifted to different extents. In Fig. 11(b), we can find that the changing rate of the wind speed was mild and the variation range was small. Nevertheless, it could also be told that the error between current prediction and the observation was smaller than other models. Compared with the yellow line (ANN) and purple line (LSTM), there were less excessive fluctuations in the blue curve. As a result, it was flattened closer to the real line. Finally in Fig. 11(c), all machine learning models showed inferior performance of R^2 on this site, while the results of persistence model were improved, showing an increase of linearity of series. However, the CNN-LSTM still showed its outperformance in several details, such as the prediction of wind speed climbing at the half and the trailing part of the series.

All of the above results and analysis indicate that the prediction from CNN-LSTM model could be better fitted with the real wind history along the timeline. Meanwhile, despite different location of the sites in wind farm, the proposed algorithm presents a satisfied robustness over other models.

4.3. Spatial comparison

Besides the better fitness in temporal series for each single site, the current CNN-LSTM model shows its specific power in providing the spatial information of the wind farm, which serves a helpful reference for the wind grid regulation.

In this section, to focus on the distribution of the wind speed over the

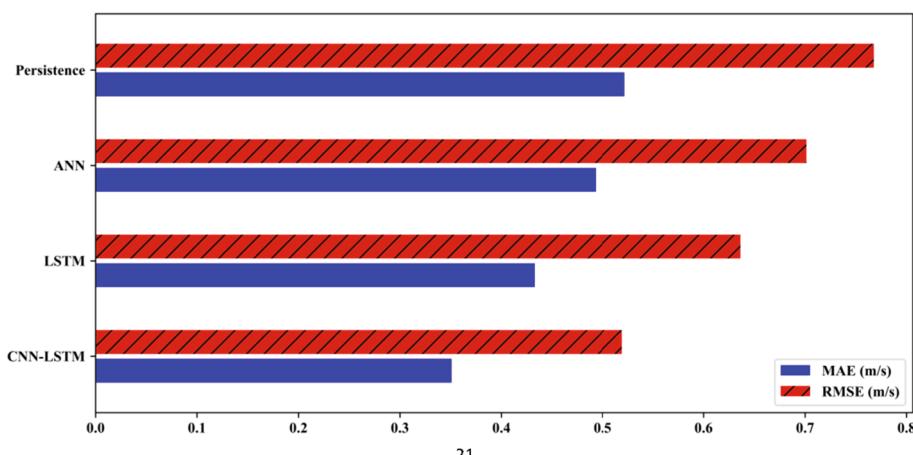


Fig. 9. Bar graph comparison of MAE and RMSE values of different models.

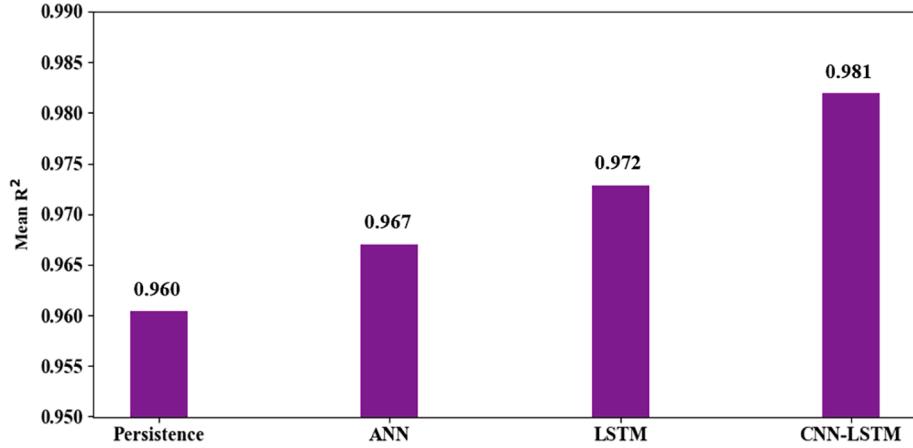


Fig. 10. Comparison of mean R^2 between prediction and observation of wind speed among sites using different models.

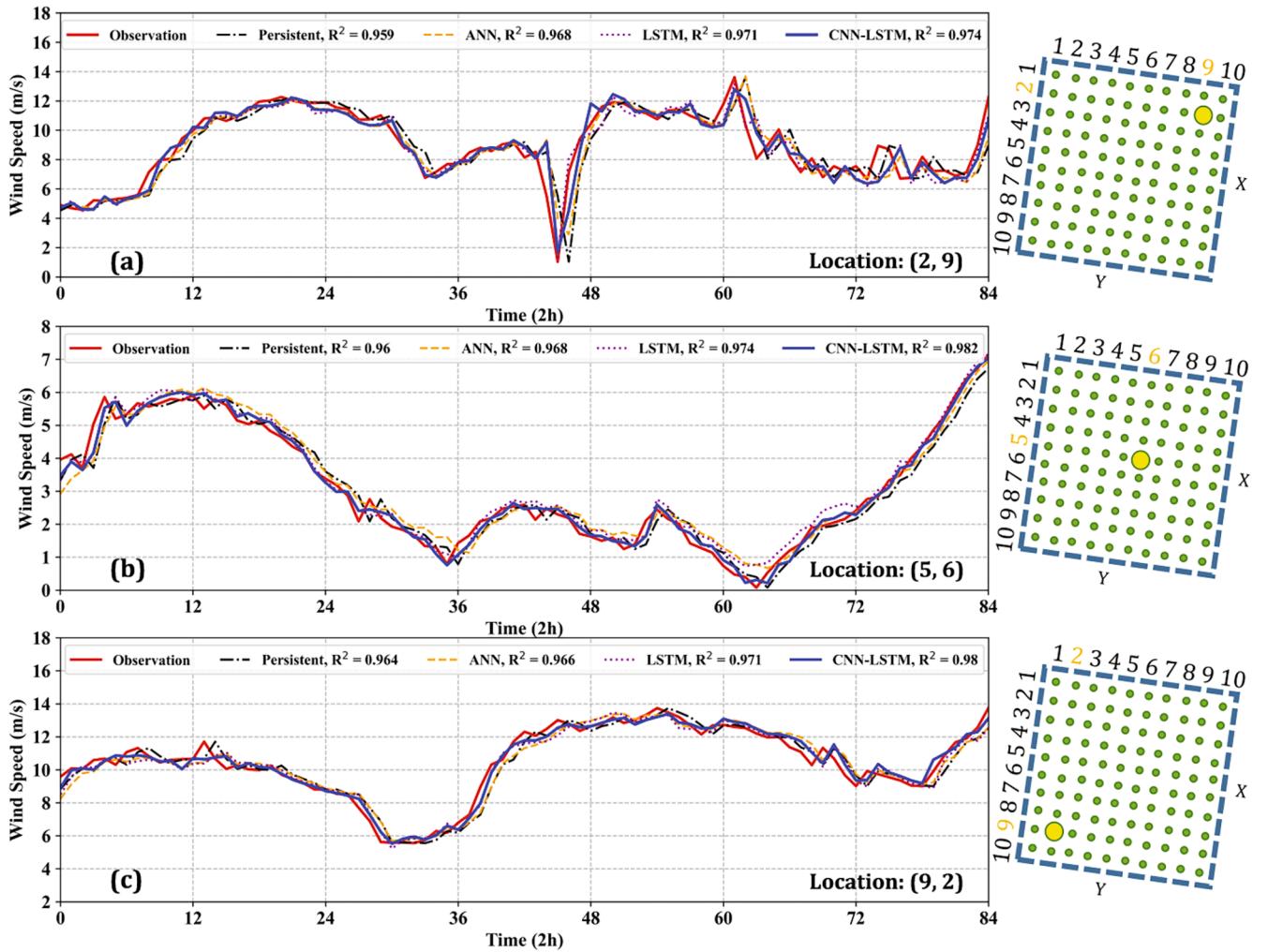


Fig. 11. Comparison of one-week extraction from testing series on different sites using different models: (a): Timestamp: 49 to 133; Location: north-east (2, 9). (b): Timestamp: 1101 to 1185; Location: middle (5, 6). (c): Timestamp: 2537 to 2621; Location: south-west (9, 2).

region, three typical kinds of timestamps from the beginning, middle and ending of the test series are selected to illustrate the superiority of current model over parallel benchmarks. The compared models include the proposed model (CNN-LSTM), LSTM, ANN and the linear model (persistence). In order to depict the wind speed contours, the wind speed heat maps are used in the following discussion, where the larger the

wind speed is, the brighter the corresponding area will be.

As shown in Fig. 12, through the comparison of the observation with persistence model, we can find that wind speed distribution was regular and it had undergone a minor change over the past 2 h. The overall wind contour had experienced a slightly east-shift from the timestamp ahead. In parallel machine learning models, the predictions had undergone

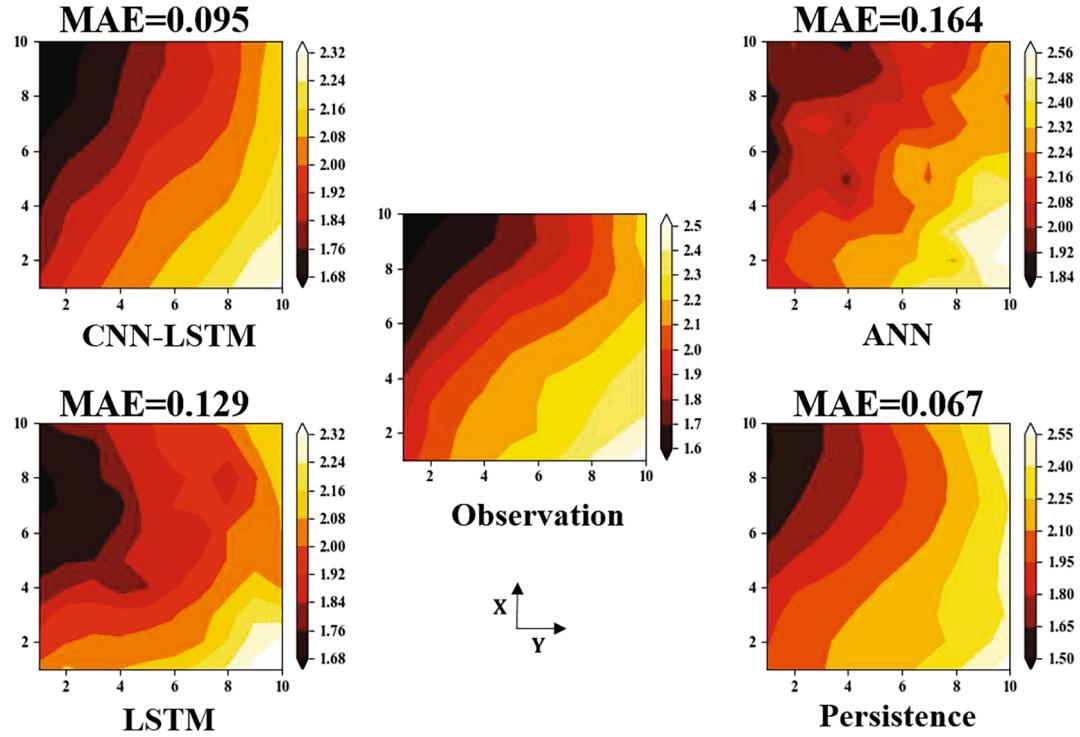


Fig. 12. Regional wind speed heat map of No. 2474 time step in test set.

“excessive learning”, where the figure of LSTM model had bent the contours and the ANN result showed an even worse distortion. However, by comparing CNN-LSTM with the observation, the proposed model had clearly kept the wind speed layer structured and successfully captured this high-linear variation trend.

Next, as shown in Fig. 13, a relatively larger change of the wind speed distribution occurred in some parts of this region. The high wind

speed zones shrank and appeared in the west-north corner of the region while the low wind speed zone also became smaller and appeared in the east-south of the wind farm. Such trend could not be well forecasted by merely using temporal model (i.e. LSTM). Moreover, in this comparison, the overall MAE of CNN-LSTM (0.164) was dramatically lower than ANN (0.422) and persistence (0.445) model, and its wind speed heat map was evidently more similar to the true label than LSTM: the

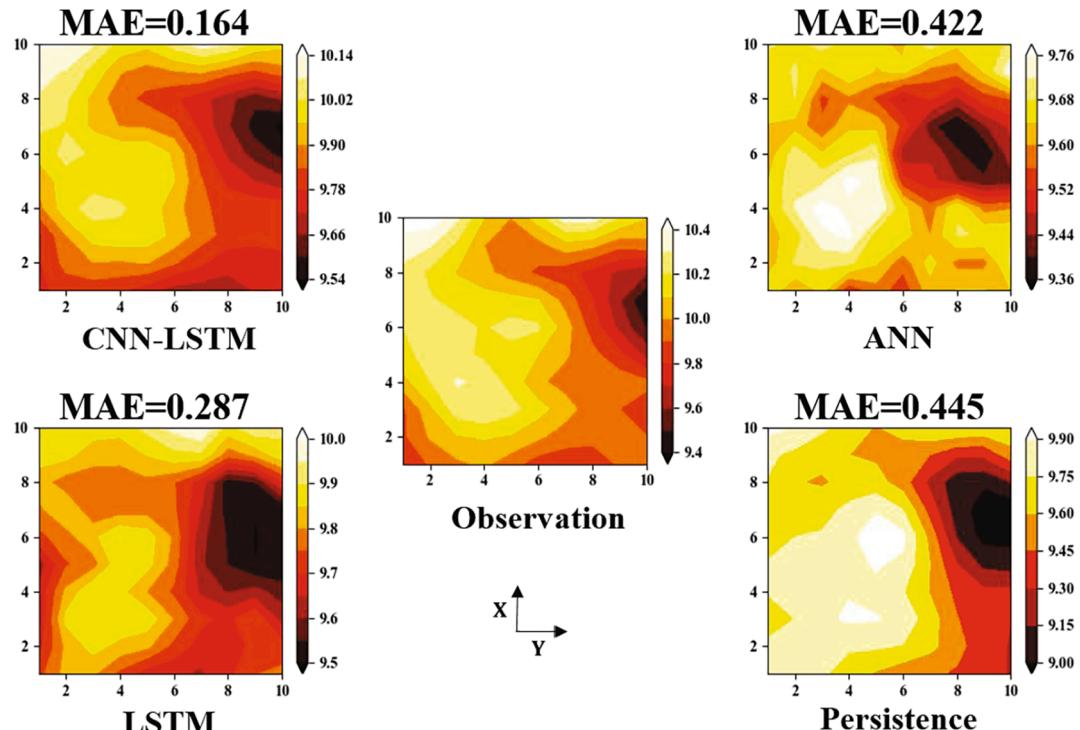


Fig. 13. Regional wind speed heat map of No. 432 time step in test set.

boundary of wind speed contour was clearer and the distribution was more consistent to the real. Such results strongly demonstrated the power of CNN filters in capturing spatial features of the wind farm.

Finally, in the third instance (Fig. 14), a totally inverted wind speed distributions are presented in the maps of observation and persistence. The low wind speed zone had moved from the north to the south part of the region while the high wind speed zone had suddenly broadcasted from the east. Such dramatic change contributed to a high loss (0.627) of persistence model. Under this circumstance, the CNN-LSTM model still well-depicted this trend, despite that the boundary of high-low speed zone was not as smooth as the observation. However, other machine learning models could hardly reflect such variation, with various fragments of wind speed zone scattered in their prediction maps and their MAE loss are considerably lower than CNN-LSTM.

All of the above comparison shows the reliability of the current model for wind speed forecasting under different wind variation conditions, indicating the involvement of CNN can greatly help to improve the stability of the estimation on regional wind speed distribution.

The reasons for the superiority of the proposed CNN-LSTM model can be considered from two aspects. First, it attributes to the “dimensionality reduction” of the auto-encoder. For the current problem of wind speed prediction regarding a 2-D regional wind farm, since the magnitude of wind speed should gradually, instead of abruptly, change between the adjacent sites, there exists multi-collinearity between the high-dimension input features. For machine learning algorithms using this input (e.g. ANN and LSTM in this work), such multi-collinearity could lead to the instability of the solution space [37], making it difficult to learn the pattern of the dataset, hence hinders the generalization of the model. However, through the auto-encoder, the high dimensional inputs can be transformed into the latent representations with much lower dimension. Such embedding vectors retain the essential characteristics of the raw input while mitigating the relevance among features. Meanwhile, its reduced dimensionality helps cut down the number of untrained parameters of the regressor, which could benefit the convergence of the training process and save the computational budget.

The second reason is that there are the better results owe to the

CNN's capability to learn the spatial structure of the provided image. The wind speeds of 2-D regional wind farm are spatially correlated. Purely using the fully-connected neural networks for prediction will lose the spatial relevance among input features, because they treat the wind speed measurements that are far away and close to each other on the same basis [38]. However, CNN can make full use of the regional information by using multiple small-sized convolutional filters to gradually extract the local distributions from the global wind speed map. Meanwhile, the pooling layer, keeping only the dominant features each time, can reduce the noise transmission [38]. As a result, the low-dimensional deep feature retains the essential spatial representation from the raw 2-D input, and when it is sent to the LSTM model to train the predictor, the predicted results will also contain the spatial information.

5. Conclusion and future work

In this work, aiming at a 2-D regional wind farm, a novel deep learning model was proposed for the short-term wind speed prediction of the site array. The model was composed of two main modules: CNN and LSTM. After the orthogonalization pre-processing, the wind speed components array was first trained by a CNN model as a structured 2D matrix, where the input and output were strictly identical and the product was the deep feature at the half-way of the network that integrates the spatial information of the wind farm. Next, through the LSTM unit, such deep feature was recurrently predicted along the timeline, using deep information from the past steps. Then, those forecasted deep features were sent back into the second-half of CNN model, known as the decoder, to constitute the 2D matrix of the wind speed components. In the end, the final prediction of the wind site array was the vector sum of the above components prediction.

The effectiveness and reliability of the proposed CNN-LSTM model have been comprehensively validated through analytical study. In terms of different assessments, including MAE, RMSE and R^2 , the current model presents considerable enhancements over the parallel mainstream models. The parallel modes are intentionally selected as

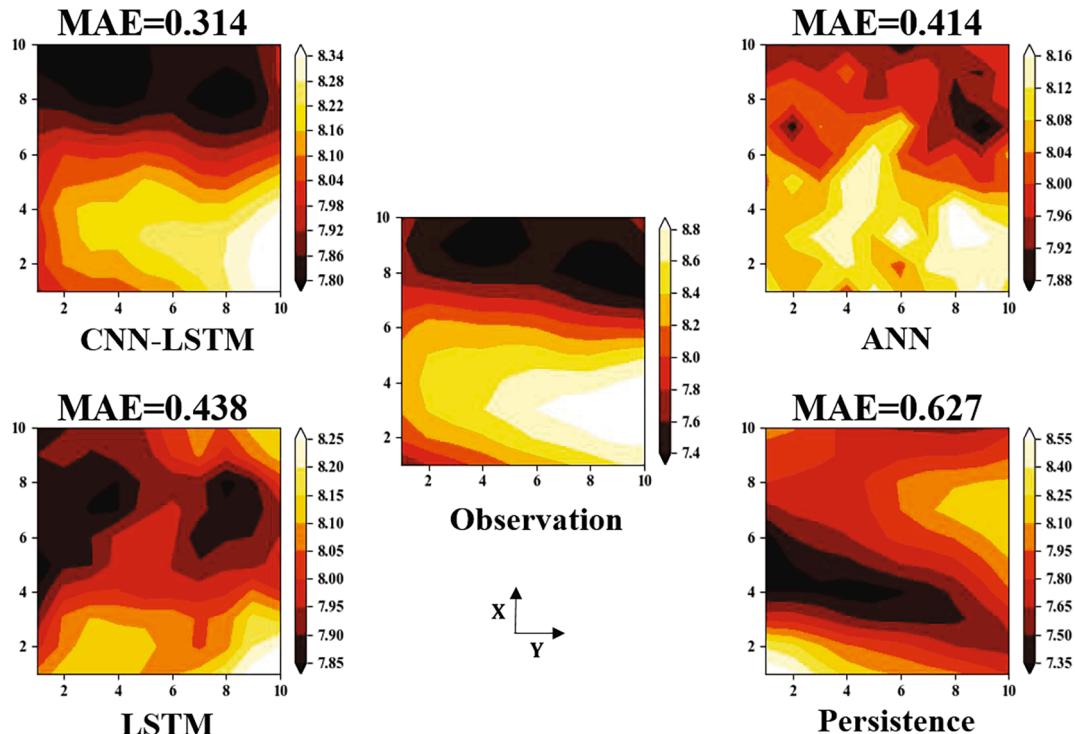


Fig. 14. Regional wind speed heat map of No. 1321 time step in test set.

persistence (representing linear model), ANN (representing naïve non-linear model) and LSTM (representing non-linear model considering temporal relations). The general improvement of MAE and RMSE are around 32%, 27% and 18% over those benchmarks, showing an impressive stepping increase.

Moreover, this study has emphasized the discussion of the current model on the prediction of the wind speed distribution of the regional wind farm. Thanks to the CNN algorithm, the model successfully forecasted the contour of wind speed zone under different scenarios, with clear and distinct boundaries between speed zones. However, there were various fragments of fake speed zones in other models and their performance were evidently lower than the proposed approach. This showed the unique power of CNN-LSTM both in temporal series forecasting and in spatial estimation. Such character would hopefully improve the resource allocation and management for the target wind farm.

In the future, the following aspects are considered to enrich the current work. Firstly, more 2-D regional wind farm datasets around different areas of the world could be added to test the generalization ability of the model; Second, various pre-processing techniques could be used and compared to further enhance the prediction accuracy; Third, criteria could be conceived to quantitatively assess the distribution of wind speed in the region and based on such criteria, a more specific algorithm for spatial wind speed prediction could be designed.

CRediT authorship contribution statement

Yaoran Chen: Investigation, Methodology, Software, Data curation. **Yan Wang:** Data curation, Methodology. **Zhikun Dong:** Methodology. **Jie Su:** Review. **Zhaolong Han:** Supervision, Review. **Dai Zhou:** Conceptualization, Supervision. **Yongsheng Zhao:** Supervision. **Yan Bao:** Supervision.

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

The financial supports from the National Natural Science Foundation of China (Nos. 42076210, 51879160, 51809170 and 11772193), Innovation Program of Shanghai Municipal Education Commission (No.:2019-01-07-00-02-E00066), and Shanghai Natural Science Foundation (No 18ZR1418000) are gratefully acknowledged. This research is also sponsored in part by the Oceanic Interdisciplinary Program of Shanghai Jiao Tong University (No.SL2020PT201), Program for Professor of Special Appointment (Eastern Scholar) at Shanghai Institutions of Higher Learning (No. TP2017013), Program for International Cooperation of Shanghai Science and Technology (No.18160744000) and “Shuguang Program” supported by Shanghai Education Development Foundation and Shanghai Municipal Education Commission (No.19SG10).

References

- [1] Hu H, Wang L, Tao R. Wind speed forecasting based on variational mode decomposition and improved echo state network. *Renewable Energy* 2021;164: 729–51.
- [2] Zhao J, Guo Z, Guo Y, Lin W, Zhu W. A self-organizing forecast of day-ahead wind speed: selective ensemble strategy based on numerical weather predictions. *Energy* 2021;218:119509.
- [3] Fu W, Zhang K, Wang K, Wen B, Fang P, Zou F. A hybrid approach for multi-step wind speed forecasting based on two-layer decomposition, improved hybrid de-hho optimization and kelm. *Renewable Energy* 2021;164:211–29.
- [4] Song D, Yang Y, Zheng S, Deng X, Yang J, Su M, et al. New perspective on maximum wind energy extraction of variable-speed wind turbines using previewed wind speeds. *Energy Convers Manage* 2020;206:112496.
- [5] Kibona TE. Application of WRF mesoscale model for prediction of wind energy resources in Tanzania. *Scientific African* 2020;7:e00302. <https://doi.org/10.1016/j.sciaf.2020.e00302>.
- [6] Prósper MA, Otero-Casal C, Fernández FC, Miguez-Macho G. Wind power forecasting for a real onshore wind farm on complex terrain using WRF high resolution simulations. *Renewable Energy* 2019;135:674–86.
- [7] Min Y, Bin W, Liang-Li Z, Xi C. Wind speed forecasting based on EEMD and ARIMA. *Chin Autom Congress (CAC)* 2015. <https://doi.org/10.1109/cac.2015.7382700>.
- [8] Khodayar M, Wang J, Manthouri M. Interval deep generative neural network for wind speed forecasting. *IEEE Trans Smart Grid* 2019;10(4):3974–89.
- [9] Moreno SR, Silva RGD, Mariani VC, Coelho LDS. Multi-step wind speed forecasting based on hybrid multi-stage decomposition model and long short-term memory neural network. *Energy Convers Manage* 2020;213:112869.
- [10] Lowery C, O’Malley M. Impact of wind forecast error statistics upon unit commitment. *IEEE Trans Sustain Energy* 2012;3(4):760–8.
- [11] Zhang Z, Ye L, Qin H, Liu Y, Wang C, Yu X, et al. Wind speed prediction method using shared weight long short-term memory network and gaussian process regression. *Appl Energy* 2019;247:270–84.
- [12] Erdem E, Shi J. ARMA based approaches for forecasting the tuple of wind speed and direction. *Appl Energy* 2011;88(4):1405–14.
- [13] Tong JL, Zhao ZB, Zhang WY. In: June. A new strategy for wind speed forecasting based on autoregression and wavelet transform. *IEEE*; 2012. p. 1–4.
- [14] Vinothkumar T, Deeba K. Hybrid wind speed prediction model based on recurrent long short-term memory neural network and support vector machine models. *Soft Comput* 2020;24(7):5345–55. <https://doi.org/10.1007/s00500-019-04292-w>.
- [15] Cai H, Jia X, Feng J, Li W, Hsu YM, Lee J. Gaussian process regression for numerical wind speed prediction enhancement. *Renewable Energy* 2020;146(2): 2112–23.
- [16] Khosravi A, Machado L, Nunes RO. Time-series prediction of wind speed using machine learning algorithms: A case study Osorio wind farm, Brazil. *Appl Energy* 2018;224:550–66.
- [17] Liu H, Mi X, Li Y. An experimental investigation of three new hybrid wind speed forecasting models using multi-decomposing strategy and elm algorithm. *Renew Energy* 2018;123:694–705.
- [18] Yan X, Liu Y, Xu Y, Jia M. Multistep forecasting for diurnal wind speed based on hybrid deep learning model with improved singular spectrum decomposition. *Energy Convers Manage* 2020;225:113456. <https://doi.org/10.1016/j.enconman.2020.113456>.
- [19] Peng Z, Peng S, Fu L, Lu B, Tang J, Wang Ke, et al. A novel deep learning ensemble model with data denoising for short-term wind speed forecasting. *Energy Convers Manage* 2020;207:112524. <https://doi.org/10.1016/j.enconman.2020.112524>.
- [20] Malik H, Yadav AK. A novel hybrid approach based on relief algorithm and fuzzy reinforcement learning approach for predicting wind speed. *Sustainable Energy Technol Assess* 2021;43:100920. <https://doi.org/10.1016/j.seta.2020.100920>.
- [21] Aly HHH. A novel deep learning intelligent clustered hybrid models for wind speed and power forecasting. *Energy* 2020;213:118773. <https://doi.org/10.1016/j.energy.2020.118773>.
- [22] Trebing K, Mehrkanon S. In: December. Wind speed prediction using multidimensional convolutional neural networks. *IEEE*; 2020. p. 713–20.
- [23] Fukukawa R, Suzuki H, Kitajima T, Kuwahara A, Yasuno T. Wind speed prediction model using LSTM and 1D-CNN. *J Signal Process* 2018;22(4):207–10.
- [24] Chen Y, Dong Z, Wang Y, Su J, Han Z, Zhou D, et al. Short-term wind speed predicting framework based on eemd-ga-lstm method under large scaled wind history. *Energy Convers Manage* 2021;227:113559. <https://doi.org/10.1016/j.enconman.2020.113559>.
- [25] Yao W, Huang P, Jia Z. In: Multidimensional LSTM networks to predict wind speed. *IEEE*; 2018. p. 7493–7.
- [26] Draxl C, Clifton A, Hodge BM, McCAA J. The wind integration national dataset (wind) toolkit. *Appl Energy* 2015;151(aug.1):355–66.
- [27] Holladay JS, LaRiviere J. The impact of cheap natural gas on marginal emissions from electricity generation and implications for energy policy. *J Environ Econ Manage* 2017;85:205–27.
- [28] Howard B, Waite M, Modi V. Current and near-term ghg emissions factors from electricity production for new york state and new york city - sciencedirect. *Appl Energy* 2017;187(Feb. 1):255–71.
- [29] Henson WLW, McGowan JG, Manwell JF. Utilizing reanalysis and synthesis datasets in wind resource characterization for large-scale wind integration. *Wind Eng* 2012;36(1):97–109.
- [30] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- [31] Yu W, Yang K, Bai Y, Xiao T, Yao H, Rui Y. June. Visualizing and comparing AlexNet and VGG using deconvolutional layers. In: Proceedings of the 33rd International Conference on Machine Learning. 2016.
- [32] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems* 2012;25:1097–105.
- [33] Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput* 1997;9(8): 1735–80.
- [34] Chen Y, Dong Z, Wang Y, Su J, Han Z, Zhou D, et al. Short-term wind speed predicting framework based on EEMD-GA-LSTM method under large scaled wind history. *Energy Convers Manage* 2021;227:113559.
- [35] Gulli A, Pal S. Deep learning with Keras. Packt Publishing Ltd.; 2017.

- [36] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
- [37] Senawi A, Wei H-L, Billings SA. A new maximum relevance-minimum multicollinearity (MRmMC) method for feature selection and ranking. Pattern Recogn 2017;67:47–61.
- [38] Nielsen MA. Neural networks and deep learning, Vol. 25. San Francisco, CA: Determination press; 2015.