



Full length article

Multi-task prediction of wind speed and time-varying wind shear coefficient using dynamic graph interactive neural network

Ke Fu, Zhengru Ren*

Institute for Ocean Engineering, Shenzhen International Graduate School, Tsinghua University, Shenzhen 518055, China

ARTICLE INFO

Keywords:

Wind speed
Wind shear
Time series prediction
Multi-task learning
Graph neural network

ABSTRACT

In the field of wind energy utilization, low-altitude wind speed and wind shear coefficient serve as pivotal variables for wind speed extrapolation, and thus, the wind speed at hub height or specific height can be effectively inferred. Although the power law model is widely used to describe wind profiles, traditional studies often assume that the wind shear coefficient is constant, typically $1/7$. This simplification ignores the dynamic changes of the wind shear coefficient and potentially lead to prediction errors. To solve this problem, this study proposed an innovative multi-task prediction method using dynamic graph interactive neural network (DGINet), and the proposed method supports parallel computing. The novelty of this study lies in fully considering the time-varying characteristics of the wind shear coefficient and can accurately predict the wind speed and wind shear coefficient at the same time, so as to more accurately construct a vertical wind profile. The proposed DGINet consists of idealized sub-networks simplified to individual neuron and backbone network adopting the encoder-decoder architecture. The proposed encoder includes modified sample reconstruction strategy within the sliding window, which expands the data dimension, and fuses the improved gated graph unit with the cross convolution operator to model and perceive the multi-level correlation between samples. The experimental results show that the proposed model can accurately and simultaneously predict wind speed and wind shear coefficient within the prediction horizons of 15 min, 30 min and 1 h.

1. Introduction

In contrast to the environmentally detrimental and exhaustible nature of fossil fuels, wind energy stands out as a favored, clean, and renewable resource [1]. Wind power generation is the main form of wind energy utilization and has grown rapidly in recent years. Notably, the energy yield from wind turbines is intricately linked to the cube of the wind speed [2]. Consequently, precise wind speed prediction is crucial for assessing wind energy potential and integration into the power grid [3].

Compared with ordinary wind speed prediction tasks, the difference is that the wind turbine hub height needs to be considered, which is usually between 50 and 150 m. Current research often relies on wind speed at the hub height or the equivalent rotor wind speed for wind power assessment [4]. Therefore, vertical wind patterns within wind farms are crucial for wind energy research, as they show how wind speed and direction change with height. However, measuring these conditions at turbine hub height requires tall towers that are costly to install [5] and maintain. From an engineering perspective, the deployment and upkeep of numerous such towers present practical inconveniences [6]. In a wind farm, a more feasible strategy involves

the establishment of a single high wind tower complemented by several lower auxiliary towers. In this way, the wind speed of the vertical wind profile is extrapolated to the desired height using the wind shear model and the lower-altitude wind speed [7]. Moreover, significant wind shear can induce vibrations or destructive loads, leading to rotor blade fatigue [8] and potentially reducing the lifespan of the wind turbine [9]. Consequently, precise prediction of the wind shear coefficient is crucial [10], as it is the only variable in the wind shear model.

In current research, power law or logarithmic law is generally used to characterize the wind shear phenomenon [11]. According to the literature summary [12,13], the power law stands out as the most dependable and widely utilized model for extrapolating wind energy, often employing a fixed empirical value of $1/7$ or specific constant. Current methodologies typically rely on such a fixed value to describe vertical wind speed profiles. However, this approach inherently assumes a constant mathematical relationship between wind speed and height, which is often an oversimplification. Consequently, this limitation, combined with the variability of actual wind patterns, can lead to significant errors when applied universally [14]. Literatures indicate that the wind shear coefficient is not static but changes dynamically

* Corresponding author.

E-mail address: zhengru.ren@sz.tsinghua.edu.cn (Z. Ren).

with wind speed at lower altitudes and various meteorological conditions [15], such as temperature gradients [16], time of day [8], and atmospheric stability [17]. Engineers use predicted wind speed measurements at lower heights along with a mathematical factor describing how wind speeds change vertically to estimate full wind patterns. Improving the accuracy of this vertical wind-speed relationship can lead to more reliable evaluations of a site's wind energy potential [18]. Table 1 provides an overview of the research pathways for wind speed extrapolation [19]. Given the interdependence of wind speed and the wind shear coefficient, both of which are critical prediction metrics, simultaneous prediction of these variables would be highly beneficial. Therefore, it is proposed to employ a multi-task learning approach to concurrently predict wind speeds and shear coefficients. Then, the problem becomes how to design a multi-task learning model to achieve simultaneous prediction of wind speed and wind shear coefficient with high accuracy.

This approach is particularly well-suited for the study's application because, in the field of wind power prediction, multi-task learning [20] has already demonstrated its effectiveness in addressing multiple related tasks simultaneously. With the help of sharing representations [21], it enhances the model's performance across each task, and the exploitation of inter-variable correlations effectively relieves the challenge of data scarcity [22–25]. Multi-task learning facilitates the joint prediction [26] of wind speed and wind power across various locations [27], taking into account spatial correlations [28]. Furthermore, for multivariate tasks, which include temperature, wind speed, and air pressure, multi-task learning can identify the correlations and interactions in the evolutionary patterns of different variables, enabling synchronous meteorological prediction [29]. Not only limited to meteorological information, the interdependencies and coupling information between wind and photovoltaic power outputs within the same region can also achieve complementary and synchronous prediction [30]. However, there is still a **knowledge gap** in the multi-task prediction of wind speed and wind shear coefficient. The limitations of existing research are not only this, but also the main approach in wind speed prediction, due to the limited number of variables available, has been to focus on single-variable prediction, and the recurrent neural networks widely used for prediction do not support parallel computing. More importantly, despite there are many studies on wind shear phenomenon, there is a notable scarcity of studies dedicated to the dynamic tracking and prediction of the wind shear coefficient.

To address this identified gap, particularly the scarcity of dynamic wind shear coefficient prediction research, this study constructs a multi-task learning model. This model simultaneously predicts wind speed and the wind shear coefficient, thereby overcoming the challenge of coupling between wind shear phenomena and wind speed. Its advantage lies in strengthening the feature expression ability of the backbone network and simplifying the sub-network of a specific task into a single neuron, avoiding the problem of task-independent modeling caused by complex sub-networks, and realizing the ideal form of multi-task learning. The model was structured around an encoder–decoder architecture. Initially, a novel sample reconstruction method within the sliding window was proposed, tailored to meet the demands of dynamic graph neural network processing. This method leverages the principles of phase space reconstruction, effectively elevating the original two-dimensional variables to a 16-dimensional space through time delay reconstruction and time series order folding techniques. On this basis, the reconstructed 16-dimensional sample features were treated as nodes, and the node features were represented by low-order polynomial coefficients. Furthermore, the graph feature encoder was constructed using improved gated graph unit, and the spatial relationship feature between reconstructed data structure of the 16-dimensional sample features was extracted using a set of cross convolution operation. Afterwards, the graph feature encoding and the spatial relationship feature encoding were integrated. In the decoder component, we have simplified the sample convolution and interaction network, eliminating

downsampling and directly employing the encoder's output features for interactive learning. Due to the reduction of information loss compared to the original structure, the modeling accuracy has been improved. Finally, the model yields time series prediction results for wind speed and wind shear coefficient through two distinct neurons, respectively. The accuracy of the proposed method was verified through real-world wind speed and wind shear coefficient, and its potential for engineering applications was demonstrated by high-precision prediction for 15 min, 30 min and 1 h.

The contributions of this paper are as follows:

- A multi-task learning method based on dynamic graph interactive neural network was proposed for the simultaneous prediction of wind speed and time-varying wind shear coefficient, which helps to describe the wind profile. The idealized multi-task learning concept was improved, information mining was mainly completed through the backbone network, and each sub-network corresponding to a specific task was simplified to a single neuron.
- A new sample reconstruction method was designed, which integrates phase space reconstruction and time series sequential folding, and expands the two-dimensional variables to 16 dimensions, taking into account the correlation of multiple spatial spans and time delay effects.
- A gated graph unit was proposed, which dynamically selects and controls the flow of information through a gating mechanism, so that it can capture complex patterns. Correspondingly, the graph feature used a feature representation method of a combination of low-order polynomial coefficients.

The rest of the paper is organized as follows: Section 2 describes the problem and the method related. Section 3 elaborates on the proposed model. Section 4 describes the experimental design and the dataset. Section 5 presents the analysis of the experimental results. Finally, Section 6 concludes the paper and discusses future research plans.

2. Preliminary and problem formulation

2.1. Wind shear model

In wind farms, since it is costly and difficult to install and maintain multiple wind towers directly at the hub height, a main wind tower is usually used in conjunction with several auxiliary wind towers. As shown in Fig. 1, the auxiliary wind towers adjacent to the wind turbines can provide more accurate wind speed data. However, the wind speed at the hub height needs to be obtained by extrapolation. The power law method is a commonly used wind speed extrapolation method, and its accuracy depends on the wind shear coefficient.

The power law model is also called the Hellman exponential law model [18]

$$v_2 = v_1 \left(\frac{h_2}{h_1} \right)^\alpha, \quad \alpha = \frac{\ln(v_2/v_1)}{\ln(h_2/h_1)}, \quad (1)$$

where v_1 and v_2 are the wind speeds at h_1 and h_2 at different heights, respectively. α is the Hellman exponential, also known as the wind shear coefficient.

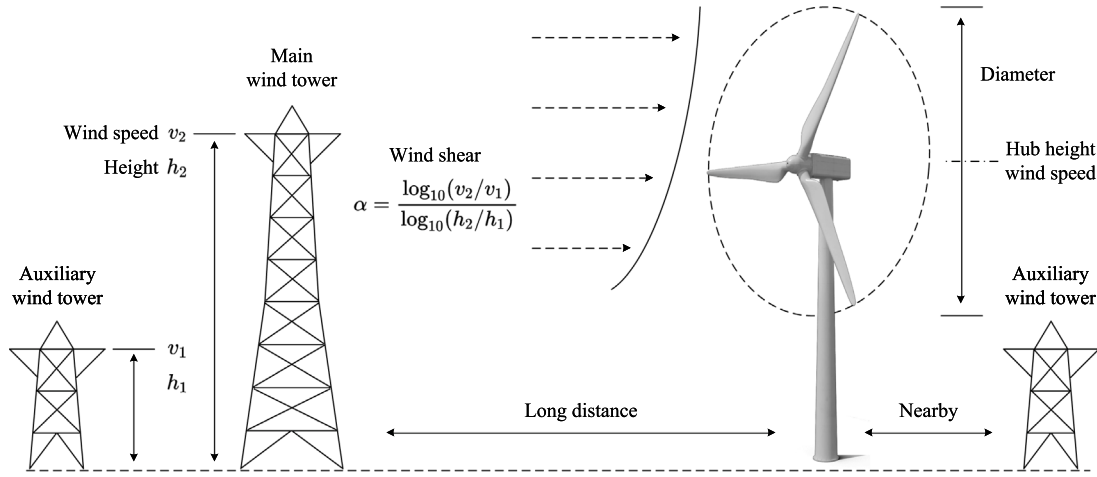
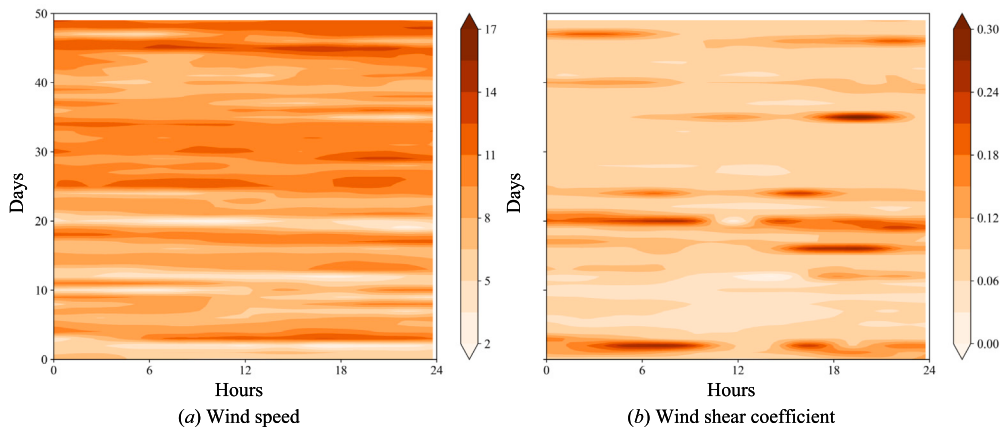
Considering that the wind shear coefficient is affected by atmospheric stability and temperature gradient, it can be assumed to be constant within the same wind farm in a short period. Based on this, combined with the measured wind shear coefficient and the wind speed at a height of 10 m, we can construct a vertical wind profile model.

Fig. 2 shows the contour map of the daily variation of wind speed and wind shear coefficient. It can be seen from the figure that the wind speed is higher around 06:00 and 18:00 every day, and the wind shear coefficient is also relatively large during these two periods. Although the difference in wind speed during the day is not large, the pattern in wind shear coefficient shows significant temporal variability, which brings challenges to accurate prediction.

Table 1

The literature review of wind speed extrapolation.

Title	Author	Source	Description
Investigation of wind shear coefficients and their effect on electrical energy generation [14]	Ebubekir Firtin (2011)	Applied energy	Wind data collected in Balıkesir from October 2008 to September 2009, has been used to show the effects of wind shear coefficient on energy production
Methods to extrapolate wind resource to the turbine hub height based on power law: A 1-h wind speed vs. Weibull distribution extrapolation comparison [11]	Giovanni Gualtieri (2012)	Renewable energy	A comparison is proposed between these two PL-based extrapolation approaches to the turbine hub height
Vertical extrapolation of wind speed using artificial neural network hybrid system [15]	Md. Saiful Islam (2017)	Neural computing and applications	Proposes two artificial neural network hybrid system-based models using genetic algorithm and particle swarm optimization for vertical extrapolation of wind speed
Use of spatio-temporal calibrated wind shear model to improve accuracy of wind resource assessment [18]	Jiale Li (2018)	Applied energy	The accuracy and performance of incorporating site-specific wind shear model calibration to predict the wind energy resource is evaluated, where six different methods are compared
A comprehensive review on wind resource extrapolation models applied in wind energy [12]	Giovanni Gualtieri (2019)	Renewable and sustainable energy reviews	A review spanning across a 40-year period (1978–2018) and has been addressed on theoretical and empirical wind resource extrapolation models applied in wind energy
A temporal model for vertical extrapolation of wind speed and wind energy assessment [13]	Paola Crippa (2021)	Applied energy	Develop a new model for wind shear coefficient which is able to capture hourly variability across a range of geographic/topographic features

**Fig. 1.** Illustration of the wind profile and distribution of wind towers at a wind farm.**Fig. 2.** Graphical representation of the contour map of the daily variation of wind speed and wind shear coefficient.

2.2. Problem formulation

Wind speed x^w and wind shear coefficient x^s are key variables for constructing wind profiles. If prediction models are established for them separately, they cannot be calculated synchronously, and the computational cost and time cost are higher than those of a unified

model. Therefore, in this paper, multi-task learning can simultaneously perform these two related prediction tasks $\{\mathcal{T}_i\}_{i=1,2}$, learn a set of functions $F(x) = \{f_i(x)\}_{i=1,2}^2$ for the two variables, and the prediction result $\hat{y}_i = F(x_i)$ can approach the actual result y_i . The multi-task loss L_{Multi} is a weighted combination of L_i , where L_i is the loss function corresponding to the prediction task.

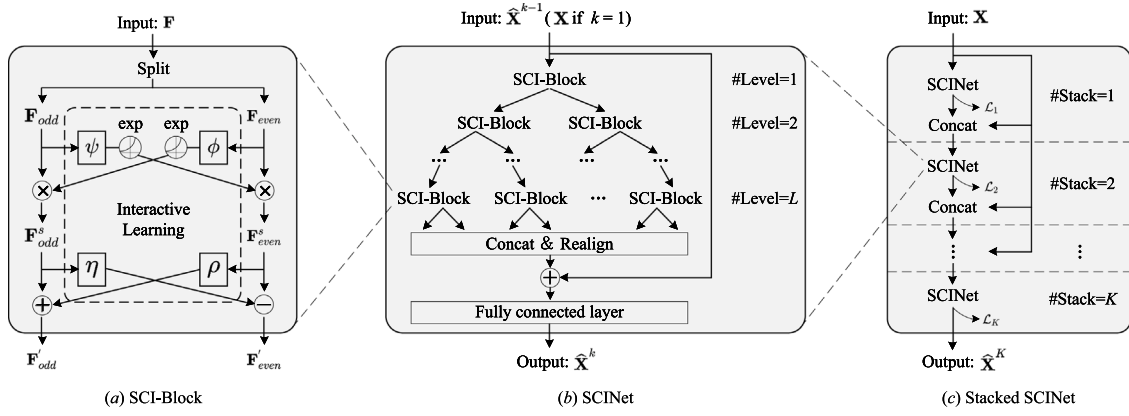


Fig. 3. Illustration of the SCINet model.

2.3. Graph neural network

As a data structure, graph can effectively model complex relationships between symbols, surpassing the representation capabilities of traditional data [31]. Whether it is a natural social network graph or the conversion of data into a graph form, such as the spatial layout of traffic flow [32], this structure can reveal deeper data connections. Therefore, graph neural networks that focus on processing dependency relationships [33] of graph-structured data are gradually being widely used in the field of natural sciences [34].

Generally, $G = (V, E, A)$ is used to represent a graph [35], where $V \in \mathbb{R}^N$ represents the nodes, E represents the edges between different nodes, $A \in \mathbb{R}^{N \times N}$ represents the adjacency matrix, and N is the number of nodes. Nodes V have two attributes, one is its own data, and the other is the feature F of nodes V . Assuming that each node collects d -dimensional features, then $F \in \mathbb{R}^{N \times d}$. Sometimes, the adjacency matrix A is replaced by the Laplacian matrix L [36], and

$$L = D - A, \quad (2)$$

$$D_{ii} = \sum_j A_{ij}. \quad (3)$$

In the above concept, nodes and edges do not change over time, which is called static graph [37]. When the graph structure or edges change over time, the dynamic graph $G(t) = (V(t_k), E(t_k), A(t_k))$ can be viewed as a series of static graphs that model spatial relationships at each time step t_k [38].

Graph convolution is suitable for modeling spatial relationships [39], and activation function σ is added to nonlinearize the graph convolutional model [40]

$$H^{(l+1)} = \sigma \left(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)} \right), \quad (4)$$

where $\tilde{A} = A + I_N$ is the adjacency matrix with added self-connections I_N , $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$, $W^{(l)}$ is trainable weight matrix. $H^{(l)} \in \mathbb{R}^{N \times D}$ is the matrix of activations in the l th layer [41].

For a two-layer graph convolutional network with a ReLU activation function, the forward model can be written as

$$Z = f(X, A) = \text{ReLU}(\hat{A} \text{ReLU}(\hat{A} X W^{(0)}) W^{(1)}), \quad (5)$$

$$\text{where } \hat{A} = \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}}.$$

2.4. SCINet

The SCINet computing framework [42] shown in Fig. 3 was proposed in 2022. As can be seen in Fig. 3(c), by stacking K layers of calculations, and the backbone architecture of the framework adopts a binary tree structure, as shown in Fig. 3(b). SCI-Block, as the core module of SCINet, is reflected in Fig. 3(a). In this module, the input feature

F is evenly divided into two sub-features F_{odd} and F_{even} through the *Splitting* operation, achieving a reduction in temporal resolution while retaining most of the original information. In order to compensate for the information loss in the downsampling process, an interactive learning strategy is adopted between the two sub-sequences, and information complementation is achieved through the *Interactive-learning* operation.

Next, different convolution kernels are applied to F_{even} and F_{odd} , respectively. Then, an *Interactive-learning* strategy is described, which achieves information interchange by learning affine transformation parameters.

Specifically, 4 different convolution operations are predefined, namely ϕ , ψ , ρ and η . F_{even} and F_{odd} are transformed to hidden states with ϕ and ψ , respectively. Further, the hidden states transformed to the formats of $\exp$, and multiply by F_{even} and F_{odd} , obtain F_{even}^s and F_{odd}^s . Here, multiply or $\times$ refers to the element-wise production.

$$F_{odd}^s = F_{odd} \times \exp(\phi(F_{even})), \quad (6)$$

$$F_{even}^s = F_{even} \times \exp(\psi(F_{odd})). \quad (7)$$

Similarly, F_{even}^s and F_{odd}^s are transformed to hidden states with ρ and η , respectively.

$$F_{odd}' = F_{odd}^s + \rho(F_{even}^s), \quad (8)$$

$$F_{even}' = F_{even}^s - \eta(F_{odd}^s). \quad (9)$$

3. Dynamic graph interactive neural network

3.1. Overall structure

In this study, Dynamic graph interactive neural network(DGINet) was proposed to solve the problem of synchronous prediction of wind speed and time-varying wind shear coefficient $\{\tau_i\}_{i=1,2}^2$, as shown in Fig. 4. The proposed model used a multi-task learning modeling framework, and the backbone network of multi-task learning was an encoder-decoder architecture.

The encoder input includes three groups, the node data matrix, the node feature matrix and the Laplacian matrix. The feature matrix and the Laplacian matrix are calculated by the proposed gated graph unit, and the data matrix is calculated by the proposed cross convolution. After that, the feature maps calculated by the two are added together to form the feature encoding of the encoder.

In the decoder, we use a simplified SCI-Block computing architecture with three differences. (1) Instead of data segmentation, we use tensor replication to assign feature codes to F_{odd} and F_{even} respectively, which avoids information loss during downsampling. (2) No $\exp$ processing is performed to simplify the calculation. (3) The final output

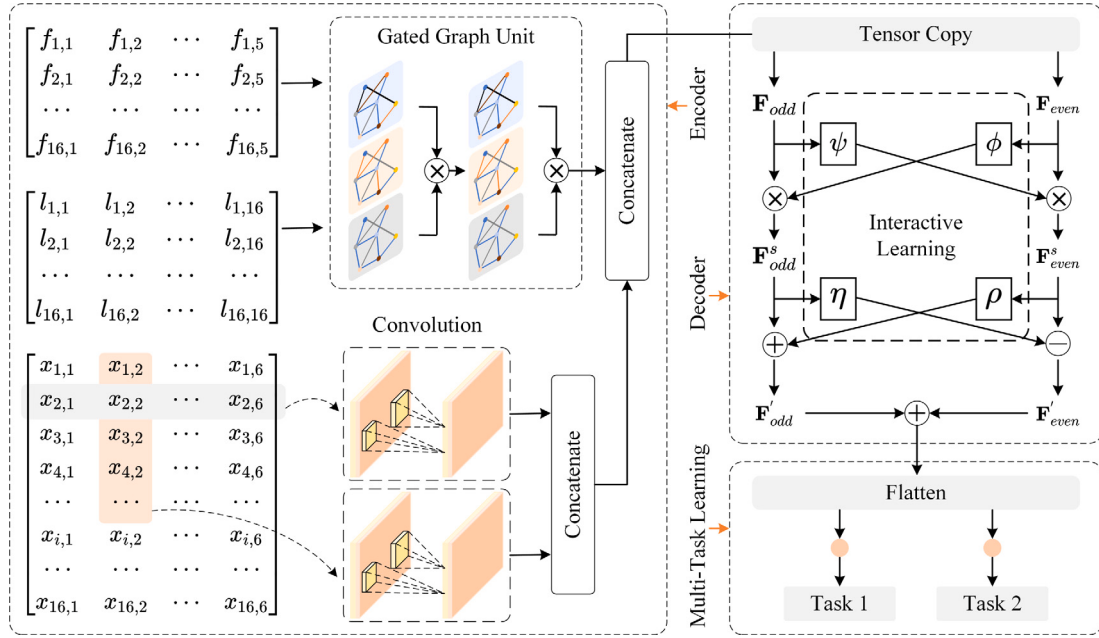


Fig. 4. Illustration of the proposed DGINet.

results F'_{odd} and F'_{even} are added. The encoder and decoder together form the backbone architecture.

The output of the backbone architecture is flattened and mapped to two different prediction tasks. Since the magnitude difference between the wind speed and wind shear coefficient studied in this paper is too large, more emphasis is placed on the wind shear coefficient in the loss function.

$$L_{Multi} = \gamma \times L_1 + \lambda \times L_2 = \gamma \times \frac{1}{n} \sum_{i=1}^n (sp_i - \widehat{sp}_i)^2 + \lambda \times \frac{1}{n} \sum_{i=1}^n (sh_i - \widehat{sh}_i)^2, \quad (10)$$

where sp_i and sh_i represent the actual wind speed and wind shear coefficient, respectively. $\widehat{sp}_i$ and $\widehat{sh}_i$ represent the predicted wind speed and wind shear coefficient, respectively. L_1 and L_2 represent the loss function of the wind speed and wind shear coefficient, respectively. $\gamma = 0.2$ and $\lambda = 0.8$ are hyperparameters used.

In addition, the Adam optimizer was used to update parameters during training.

3.2. The proposed sample reconstruction method based on phase space reconstruction

This study shows that the delay time τ of wind speed and wind shear coefficient calculated by the C-C method is 8, and the maximal Lyapunov exponent λ of both are greater than 0, 0.042 and 0.037 respectively, which confirms that they are both chaotic time series. Considering the limitation of sliding window length, in order to avoid very low sampling rate, we choose $\tau = 4$ as the delay time for phase space reconstruction. Packard's coordinate delay phase space reconstruction method and Takens theorem both support this choice, pointing out that the choice of delay time τ will not affect the characterization of the system's dynamic properties. Taking wind speed as example, we use the coordinate delay method to reconstruct the phase space to show the global characteristics of the time series, recorded as $\vec{x}^v$, and process the time series in the sliding window by the 4-equal folding method to characterize the local characteristics, recorded as x^v .

$$\vec{x}^v = \begin{bmatrix} \vec{x}_1^v & \vec{x}_{1+\tau}^v & \dots & \vec{x}_{1+(m-1)\tau}^v \\ \vec{x}_2^v & \vec{x}_{2+\tau}^v & \dots & \vec{x}_{2+(m-1)\tau}^v \\ \vec{x}_3^v & \vec{x}_{3+\tau}^v & \dots & \vec{x}_{3+(m-1)\tau}^v \\ \vec{x}_4^v & \vec{x}_{4+\tau}^v & \dots & \vec{x}_{4+(m-1)\tau}^v \end{bmatrix},$$

$$x^v = \begin{bmatrix} x_1^v & x_2^v & \dots & x_{n/4}^v \\ x_{1+n/4}^v & x_{2+n/4}^v & \dots & x_{n/2}^v \\ x_{1+n/2}^v & x_{2+n/2}^v & \dots & x_{3n/4}^v \\ x_{1+3n/4}^v & x_{2+3n/4}^v & \dots & x_n^v \end{bmatrix}. \quad (11)$$

The two dimensions of wind speed and wind shear coefficient are expanded to 16 dimensions to reveal the dynamic characteristics of wind speed and wind shear coefficient.

$$\begin{bmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,6} \\ x_{2,1} & x_{2,2} & \dots & x_{2,6} \\ \dots & \dots & \dots & \dots \\ x_{16,1} & x_{16,2} & \dots & x_{16,6} \end{bmatrix} = \text{stack}[\vec{x}^v, x^v, \vec{x}^a, x^a]. \quad (12)$$

Fig. 5 shows the characteristics of the reconstructed data through recurrence plots, where Fig. 5(a)–(d) are based on the time-delay phase space reconstruction method, while Fig. 5(e)–(h) use the sequential folding method. The color bar on the right indicates the degree of similarity, from dark purple (0) to yellow (12). The higher the brightness, the stronger the similarity of the data points. The bright lines and patches in the figure reveal the high-similarity areas, while the length of the diagonal indicates the degree of divergence of the trajectory. Random signals usually do not form long diagonals. The comparison of these recurrence plots characterizes the degree of pattern similarity of the signals. It can be observed that the sequence reconstructed in phase space Fig. 5(a)–(d) maintains the original pattern in the recurrence plot, while the sequence folded sequentially Fig. 5(e)–(h) mainly shows local features. The two methods effectively integrate the local and global features of the data.

3.3. Graph structure

Node features construction According to the proposed sample reconstruction method, the length of each row vector is 1/4 of the sliding window size. Taking the sliding window 24 used in this paper as an example, the length of each row vector after sample reconstruction is 6 samples. Thus, in order to prevent overfitting, 5 coefficients of first-order and second-order polynomials are used as data features to

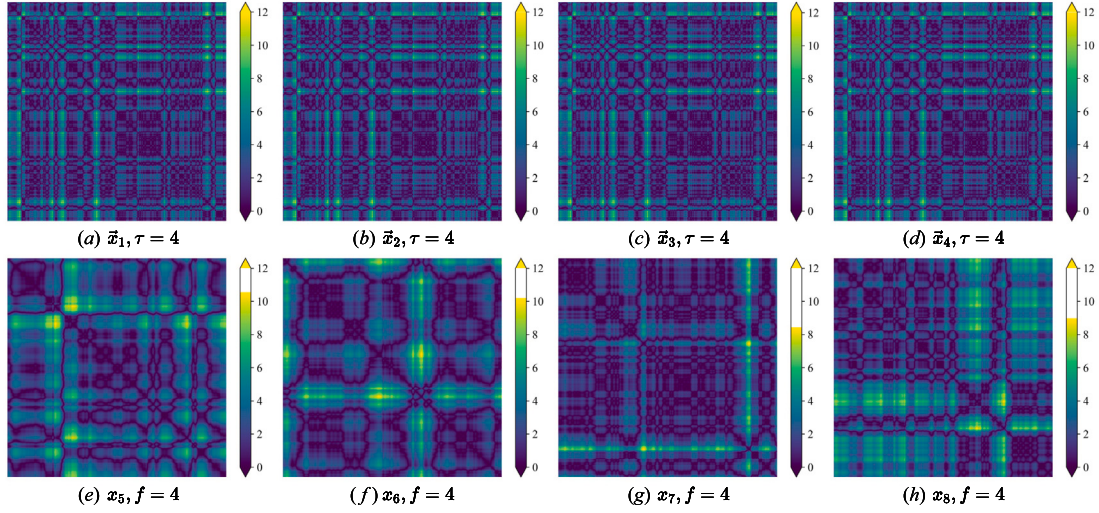


Fig. 5. Graphical representation of the characteristics of the reconstructed data through recurrence plots. (a)–(d) represent the data pattern after phase space reconstruction, while (e)–(h) represent the data pattern after sequential folding.

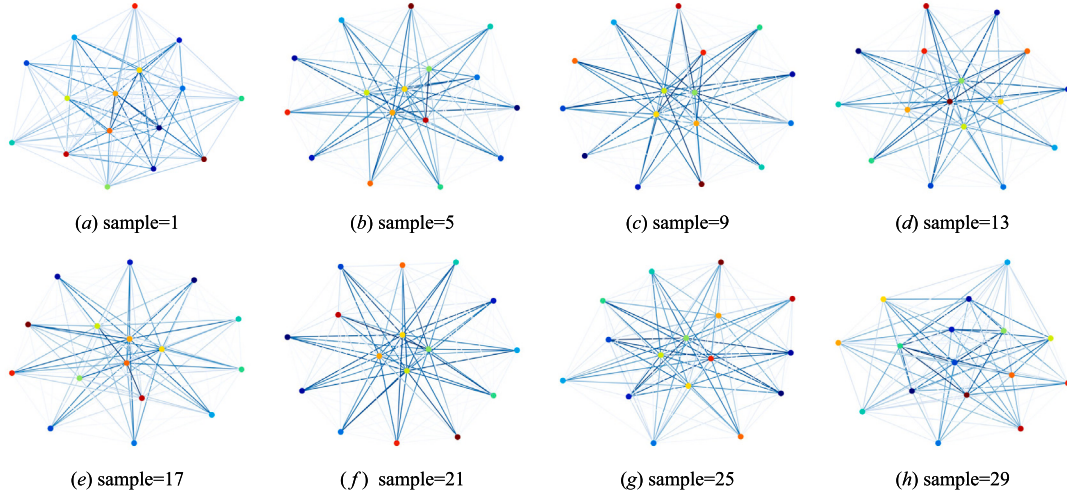


Fig. 6. Graphical representation of multiple graph structure data built based on a sliding window, which contains 16 nodes and their connecting edges. The number sample represents the position of the sliding window.

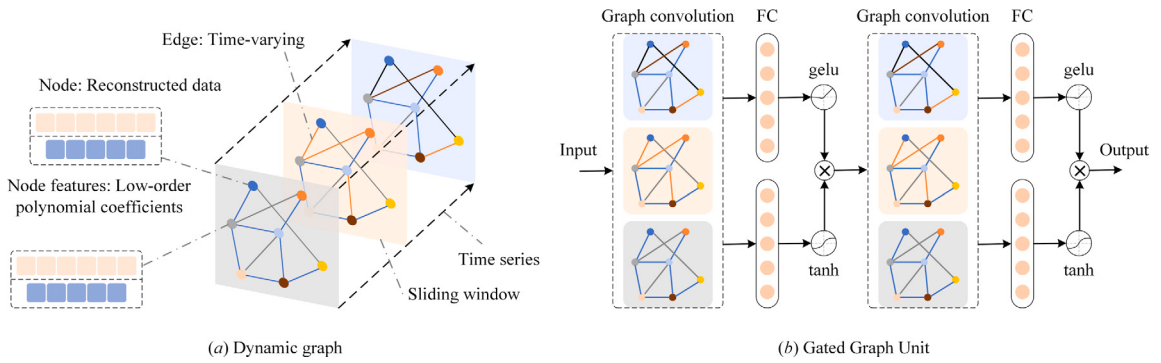


Fig. 7. Illustration of the dynamic graph and proposed gated graph unit.

simplify the model and maintain its generalization ability.

$$F = \begin{bmatrix} f_{1,1} & f_{1,2} & f_{1,3} & f_{1,4} & f_{1,5} \\ \dots & \dots & \dots & \dots & \dots \\ f_{r,1} & f_{r,2} & f_{r,3} & f_{r,4} & f_{r,5} \\ \dots & \dots & \dots & \dots & \dots \\ f_{16,1} & f_{16,2} & f_{16,3} & f_{16,4} & f_{16,5} \end{bmatrix}, r \in \{1, 2, \dots, 16\}. \quad (13)$$

The basic idea of polynomial fitting is to fit a set of data points by the least squares method to find the polynomial function $p^{(n)}(x)$.

$$p^{(n)}(x) = a_0^{(n)} + a_1^{(n)}x + \dots + a_n^{(n)}x^n, \quad (14)$$

where $a_0^{(n)}, a_1^{(n)}, \dots, a_n^{(n)}$ are the polynomial coefficients with the highest order term is x^n . The polynomial coefficients of $[f_{j,1} \ f_{j,2} \ f_{j,3} \ f_{j,4} \ f_{j,5}]$ are fitted respectively, denoted as $p^{(j,n)}(x)$. Then, the feature is calculated as

$$[f_{j,1} \ \dots \ f_{j,5}] = \begin{bmatrix} a_0^{(j,1)} & a_1^{(j,1)} & a_0^{(j,2)} & a_1^{(j,2)} & a_2^{(j,2)} \end{bmatrix}, \quad j \in \{1, 2, \dots, 16\}. \quad (15)$$

Laplace matrix construction Based on the above analysis, we constructed 16 variables in the sliding window and calculated the adjacency matrix A between them, where each element represents the Pearson correlation coefficient between the corresponding variables. Then, the Laplace matrix L is calculated using the adjacency matrix A.

In the adjacency matrix used in this paper, the elements at corresponding positions use the correlation between pairs, which is measured by the Pearson correlation coefficient. This method does not require the use of prior knowledge to predefine the adjacency matrix, and it can be automatically constructed using the data in each sliding window.

Based on the node features and Laplace matrix, it is converted into graph structure data from time series. Fig. 6 shows multiple graph structure data built based on a sliding window, which contains 16 nodes and their connecting edges. The node color is used to distinguish different nodes, while the color depth of the edge indicates the strength of the correlation between nodes. The graph contains 8 subgraphs, each of which represents a sliding window with an interval of 4 data points. From the changes in nodes and edges, it can be seen that although the graph structure remains unchanged, the connection properties of the edges change dynamically over time. It is worth noting that dynamically changing edges can better capture the complex relationships between nodes that change over time and improve the expressiveness of the model.

3.4. Gated graph unit

Gated linear units couples linear elements to gates, preserving the ability of nonlinear learning while enabling gradients to propagate across linear units [43]. Furthermore, the effectiveness of multiple variants of gated linear units was demonstrated [44]. This study uses a combination of graph convolution and gating mechanism, as shown in the Fig. 7. We compute the hidden layers h as

$$h(\mathbf{Z}) = \text{GeLU}(\mathbf{Z}\mathbf{W} + \mathbf{b}) \times \tanh(\mathbf{Z}\mathbf{V} + \mathbf{c}), \quad (16)$$

where $\mathbf{Z}$ is the result of graph convolution, $\mathbf{W}$, $\mathbf{b}$, $\mathbf{V}$, $\mathbf{c}$ are the learned parameters, GeLU and tanh are the activation function and $\times$ is the element-wise product. It is worth noting that the highest-order term in our feature matrix is set to coincide with the order of the gating mechanism.

3.5. Cross convolution operation

The cross convolution operation is shown in the lower left corner of Fig. 4, and is formulated as

$$C^{\text{cross}} = \text{Concat}([\text{ReLU}(C^S), \text{ReLU}(C^T)]), \quad (17)$$

where C is convolution operation, C^S represents convolution across variables to capture spatial dependencies, while C^T represents within-variable convolution to capture temporal dependencies. The vertically crossed C^S and C^T are stacked as the feature map of data after sample reconstruction.

4. Experiment

4.1. Experimental configuration

The experiments were implemented in Tensorflow 2.10.0 and Python 3.12, the hardware used includes Intel i7-13700k CPU, Nvidia Quadro A2000 GPU with CUDA 11.2 and 32 GB RAM. The number of channels and the size of the sliding window are both set to 24.

This baselines compared include not only classic models [45,46] such as the LSTM, gate recurrent unit (GRU) [47], temporal convolutional network (TCN), BiLSTM, BiGRU, ConvLSTM and Transformer, but also recently proposed Fnet, Fastformer, SCINet and Mamba.

4.2. Description of dataset

The dataset used comes from a 48 MW wind farm at the National Offshore Wind Power Research and Test Base in Fujian, China. The data were collected from January 3 to February 21, 2022, for a total of 50 days, and the ratios of the training set, validation set, and test set were divided in order as 0.7, 0.1, and 0.2. Specifically, we used the measured wind speed at a height of 10 m and the calculated wind shear coefficient. Since this paper adopts a multi-task learning model, both the input variables and the output variables include wind speed and wind shear coefficient. The data were smoothed using the Savitzky–Golay filter to reflect the data pattern and ignore unnecessary fluctuations. Since the climate system is a typical nonlinear system with chaotic characteristics, this paper identifies it, and the specific results are reflected in the sample reconstruction method.

4.3. Ablation experiments

The ablation experiment calculated 4 groups in total, which are divided into two categories, one containing graph structure and sample reconstruction, and the other containing only sample reconstruction without graph structure. Specifically, in the first category, the decoder is replaced by GRU and LSTM respectively, labeled graph2 and graph3, in the second category, only the graph structure is removed, the decoder is unchanged, labeled graph4, and the graph structure is removed and the decoder is replaced by LSTM, labeled graph5.

4.4. Performance metrics

Three evaluation metrics were used to measure the accuracy of model prediction, including mean absolute error (MAE), root mean square error (RMSE), and coefficient of determination (R^2), which robust to outliers, penalizes large errors, explains variance proportion, respectively. The following formulas are given

$$\begin{aligned} \text{MAE} &= \frac{1}{n} \sum_{i=1}^n |t_i - t'_i|, \quad \text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (t_i - t'_i)^2}, \\ R^2 &= 1 - \frac{\sum_{i=1}^n (t_i - t'_i)^2}{\sum_{i=1}^n (t_i - \bar{t})^2}, \end{aligned} \quad (18)$$

where t_i and t'_i represent the true value and the predicted value of the time series, respectively. Besides, n is the length of the time series, and $\bar{t}$ is the mean value of the time series. The values of MAE and RMSE are close to 0, and the values of R^2 are close to 1, indicating that the model has high accuracy. In addition, the training and inference times of the proposed model are not the longest and within a reasonable range, so they are not discussed later.

Table 2

Wind speed prediction results for different time spans (including 15 min, 30 min and 1 h).

Model	Time span								
	15 min			30 min			1 h		
	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2
LSTM	0.144275	0.186662	0.991963	0.202700	0.261226	0.984232	0.331260	0.426524	0.957851
GRU	0.120642	0.155532	0.994420	0.186978	0.240395	0.986647	0.307549	0.397504	0.963391
TCN	0.191166	0.252493	0.985294	0.220449	0.290490	0.980501	0.359098	0.455913	0.951842
BiLSTM	0.199496	0.249583	0.985631	0.257991	0.324647	0.975646	0.385021	0.490120	0.944344
BiGRU	0.143399	0.183149	0.992262	0.201654	0.256771	0.984765	0.338409	0.429407	0.957279
ConvLSTM	0.167552	0.218774	0.988959	0.224041	0.293215	0.980134	0.355232	0.460609	0.950845
Transformer	0.252241	0.321043	0.976224	0.313544	0.399596	0.963103	0.437464	0.560159	0.927301
Fnet	0.255634	0.495122	0.943450	0.297001	0.408075	0.961521	0.426793	0.551388	0.929560
SCINet	2.108964	2.580756	-0.536384	2.099604	2.570542	-0.526839	2.100183	2.571625	-0.532213
Fastformer	0.151607	0.195878	0.991149	0.208779	0.270208	0.983129	0.332696	0.430194	0.957122
Mamba	0.251698	0.321284	0.976189	0.311709	0.397990	0.963399	0.435364	0.556203	0.928325
DGINet	0.051300	0.065637	0.999006	0.091662	0.116207	0.996880	0.170763	0.220009	0.988785

5. Results and discussion

5.1. Wind speed prediction results for different time spans

Table 2 shows the difference in wind speed prediction performance between the proposed model and various baseline models at multiple time spans (including 15 min, 30 min and 1 h). In the baseline models, GRU stably achieved the best results in all three scenarios. In addition, for the suboptimal prediction results, BiGRU has the acceptable prediction performance in the 15-min and 30-min prediction scenarios. When the time span is increased to 1 h, things change and LSTM has a slightly better prediction accuracy. Nevertheless, the proposed model is far ahead in the three prediction indicators of MAE, RMSE and R^2 . Compared with the best results among various baseline models, the MAE of the proposed model decreased by 57.48%, 50.98%, 44.48% under the three prediction time spans. The RMSE decreased by 57.80%, 51.66%, 44.65%, as for R^2 , and the deviation was reduced by 82.19%, 76.63%, and 69.37% relative to the perfect prediction.

Fig. 8 illustrates the comparison between the predicted trajectory and the true value of the wind speed time series. The three rows in the figure show the prediction results for 15 min (figure (a1)-(a4)), 30 min (figure (b1)-(b4)) and 1 h (figure (c1)-(c4)). The proposed DGINet is compared with three baseline models, including GRU (a type of RNN), Fastformer (an attention model with linear computational complexity) and Mamba (a state-space model), furthermore, these three models have relatively the best prediction performance in their respective fields. All models perform well in 15-min prediction, but the accuracy decreases as the prediction time span increases, which is attributed to the increase in uncertainty. In particular, in the 1-h prediction, the prediction errors of DGINet, GRU, Fastformer and Mamba increase successively, showing more and more obvious burrs.

Fig. 9 shows the scatter plot of the wind speed time series prediction and true values, with the same layout as the prediction trajectory comparison figure. Ideally, the scatter points should be closely distributed along the ideal fitting line represented by the black solid line. In practice, the scatter points are distributed around the line, and the degree of deviation reflects the accuracy of the prediction. The scatter points in the figure are evenly distributed on both sides of the fitting line, indicating that there is no significant deviation between the predicted mean and the actual mean. As the prediction time span increases, the prediction performance of each model decreases. However, the horizontal comparison shows that the prediction value of the proposed model is closest to the ideal fitting line, indicating that its prediction effect is the best.

Fig. 10 shows the wavelet time-frequency analysis of the wind speed time series prediction error. The horizontal axis represents the length of the time series, the vertical axis represents the frequency, and the color depth corresponds to the energy intensity, blue is weak, and red is strong. As time goes on, the prediction error increases, but the

error energy of the proposed model is lower than that of the three comparison models in each time span. The analysis of the wavelet time-frequency diagram not only reveals the changes in the prediction error, but also points out that the low-frequency error implies that there is room for improvement in pattern extraction, while the high-frequency error indicates the randomness of the time series, confirming that the model accurately captures the evolution pattern of the time series.

5.2. Wind shear coefficient prediction results for different time spans

Table 3 describes the performance differences of the proposed model and the baseline models in wind shear coefficient prediction under multiple time spans (including 15 min, 30 min, and 1 h). Among the baseline models, similar to the wind speed prediction results, GRU stably achieved the best results in all three scenarios. In addition, Fastformer achieved suboptimal results in the 15-min and 30-min prediction scenarios, which is not in harmony with the wind speed prediction. When the time span increases to 1 h, LSTM has slightly better prediction accuracy. Nevertheless, the proposed model has the lowest MAE, RMSE and the highest R^2 . Compared with the best results among the baseline models, the MAE of the proposed model in the three prediction time spans decreased by 32.77%, 33.15%, and 25.65%, respectively. The RMSE decreased by 30.59%, 27.25%, and 15.19%, respectively, and the deviations were reduced by 51.82%, 47.07%, and 28.09%, respectively, relative to the perfect prediction. In addition, note that the prediction results of wind shear coefficient should be combined with the wind speed prediction results because they are jointly inferred in the multi-task learning model.

Fig. 11 shows the comparison between the predicted trajectory of the wind shear coefficient and the true value. The layout of each subgraph is consistent with the wind speed prediction subgraph. In the 15-min and 30-min prediction, it can be seen that the prediction results of Mamba have shown obvious deviations, while the predicted values of other models are almost consistent with the actual values. As the time span increases, in the 1-h prediction, the prediction effect deteriorates further, and all models have different degrees of burrs. The proposed model is more consistent, followed by GRU and Fastformer, and Mamba has the largest error.

Fig. 12 is the scatter plot of the predicted value and the true value of the wind shear time series, and the layout is the same as the prediction trajectory comparison figure. The scatter points in the figure are distributed on both sides of the fitting line instead of concentrated on one side, indicating that there is no obvious deviation between the predicted mean and the actual mean. A notable phenomenon is that most of the scatter points are concentrated in the lower left corner. This is because the value of the wind shear coefficient is generally small, and the interval of the wind shear coefficient increase is an unconventional situation. In addition, another phenomenon is that the scatter points in the lower left corner are relatively concentrated, while the upper right

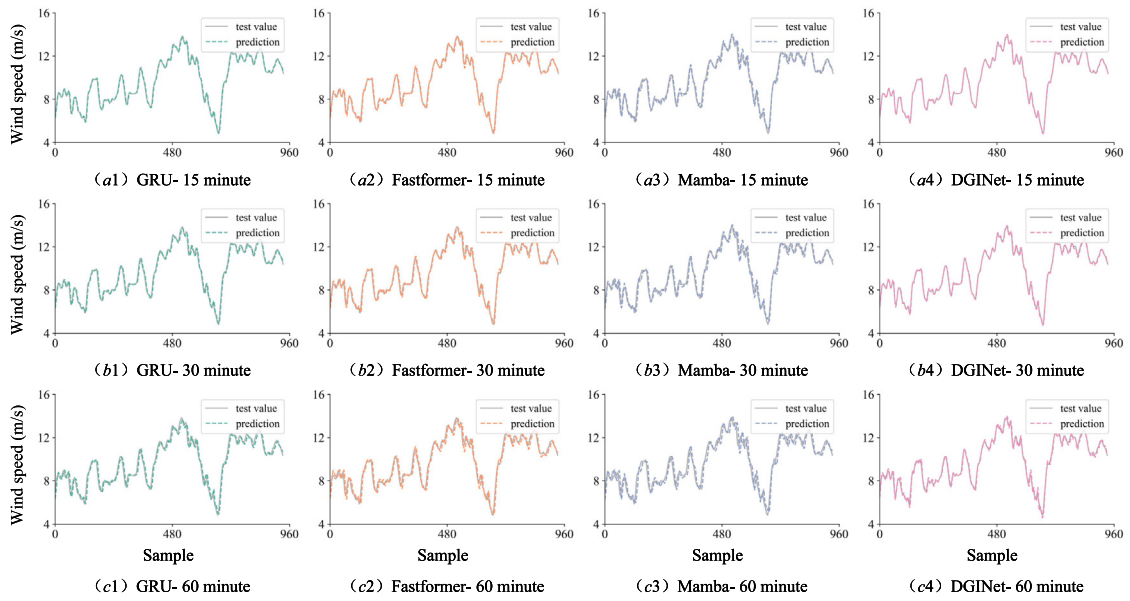


Fig. 8. Comparison of wind speed prediction track and actual test value.

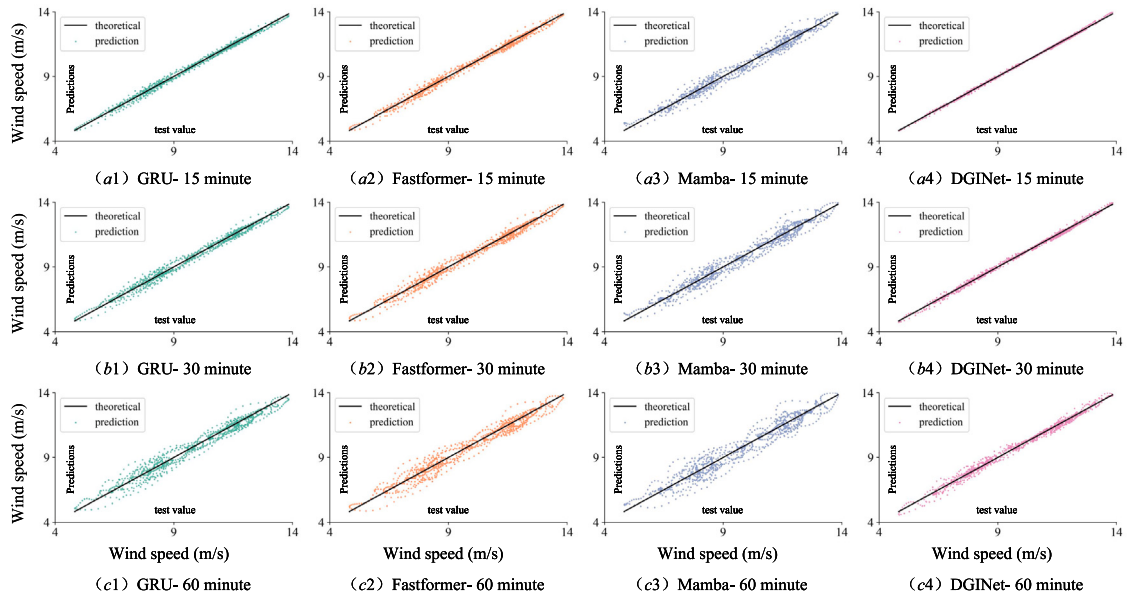


Fig. 9. Scatter plot comparison between prediction track and actual test value of wind speed.

Table 3

Wind shear coefficient prediction results for different time spans (including 15 min, 30 min and 1 h).

Model	Time span								
	15 min			30 min			1 h		
	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2
LSTM	0.001676	0.002683	0.989495	0.002458	0.003944	0.977247	0.004265	0.006930	0.929456
GRU	0.001245	0.001837	0.995077	0.001958	0.003090	0.986034	0.003945	0.006318	0.941365
TCN	0.004061	0.005218	0.960272	0.003791	0.005192	0.960564	0.006485	0.008851	0.884931
BiLSTM	0.002066	0.002989	0.986964	0.002769	0.004072	0.975739	0.004602	0.006920	0.929673
BiGRU	0.001659	0.002505	0.990847	0.002512	0.003913	0.977607	0.004407	0.006907	0.929938
ConvLSTM	0.002163	0.003011	0.986771	0.002902	0.004351	0.972309	0.004831	0.007729	0.912251
Transformer	0.005638	0.007492	0.918099	0.006329	0.008559	0.892849	0.007902	0.010899	0.825525
Fnet	0.012801	0.023488	0.194981	0.012814	0.022177	0.280560	0.013869	0.023255	0.205669
Fastformer	0.001588	0.002370	0.991806	0.002364	0.003611	0.980927	0.004277	0.006540	0.937186
SCINet	0.007999	0.009788	0.860212	0.008679	0.010604	0.835502	0.009279	0.011567	0.803483
Mamba	0.005645	0.007451	0.918990	0.006281	0.008442	0.895743	0.007645	0.010588	0.835350
DGINet	0.000837	0.001275	0.997628	0.001309	0.002248	0.992608	0.002933	0.005358	0.957835

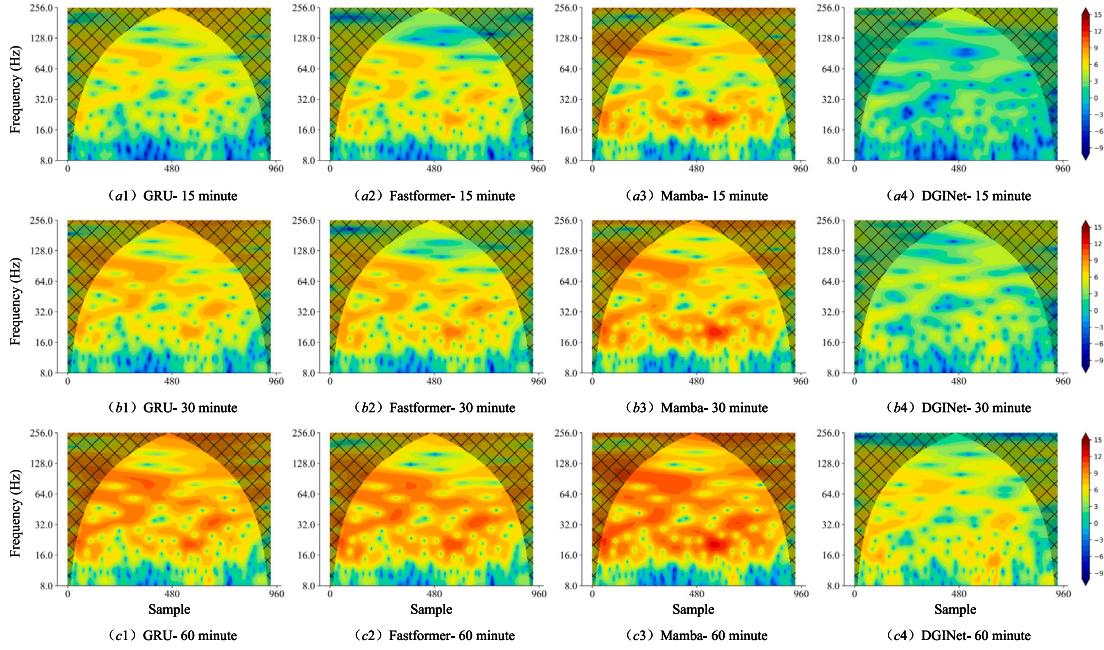


Fig. 10. Comparison of wavelet time-frequency graph of wind speed prediction errors.

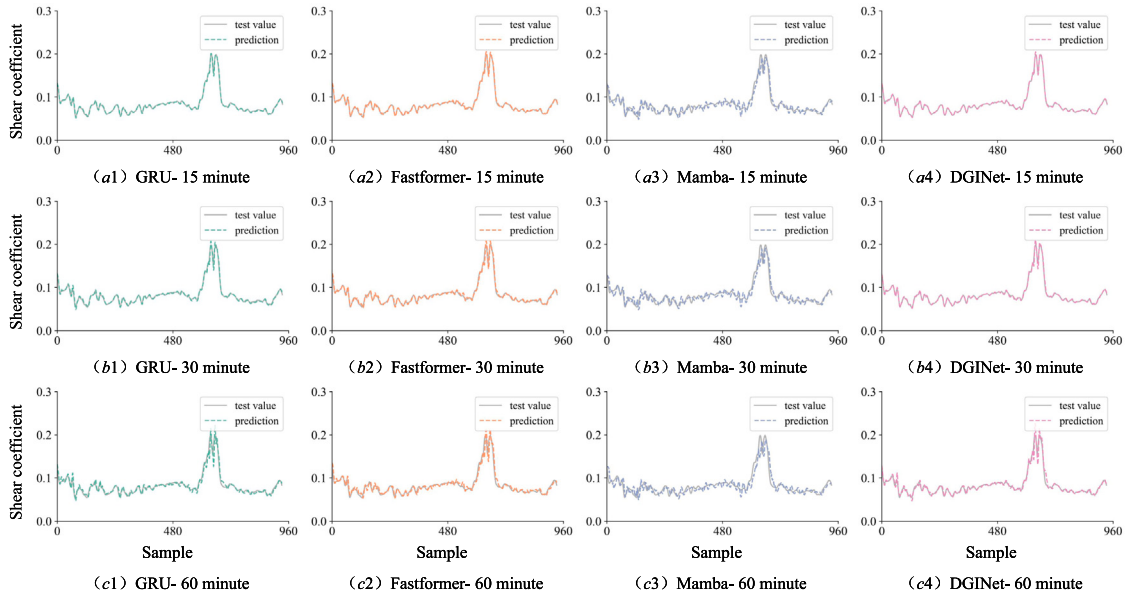


Fig. 11. Comparison of wind shear coefficient prediction track and actual test value.

corner is relatively scattered. As the prediction time span increases, this phenomenon becomes more and more significant. This is because when the wind shear coefficient is stable, the uncertainty is small, which conforms to the long-term law, while the uncertainty is large during the time period when the wind shear coefficient changes dramatically.

Fig. 13 shows the results of the wavelet time-frequency analysis of the wind shear coefficient prediction error. The representation of its graphical elements is consistent with the wavelet time-frequency figure of wind speed. As time goes by, the prediction error continues to increase, but the error energy of the proposed model is lower than that of the three comparison models in each time span. It can be seen that the error energy of Mamba is significantly higher than others, which is consistent with the prediction trajectory and more intuitive. In the wavelet time-frequency figure of the proposed model, the error energy in the high-frequency interval is significantly lower than others, which

shows that the proposed model extracts the data pattern and reduces the randomness of the prediction error. In addition, there is a high-energy area of prediction error on the right side of each subfigure, which is consistent with the interval of large changes in the wind shear coefficient. This means that when the uncertainty of the wind shear coefficient increases, the prediction error will increase accordingly.

5.3. Ablation experiment of wind speed and wind shear coefficient

Table 4 shows the difference in wind speed prediction performance between the ablation experiments at multiple time spans (including 15 min, 30 min and 1 h). In the ablation experiments, the proposed model performs best on all time spans, with the lowest MAE and RMSE and the highest R^2 value. When predicting for 15 min and 1 h, the

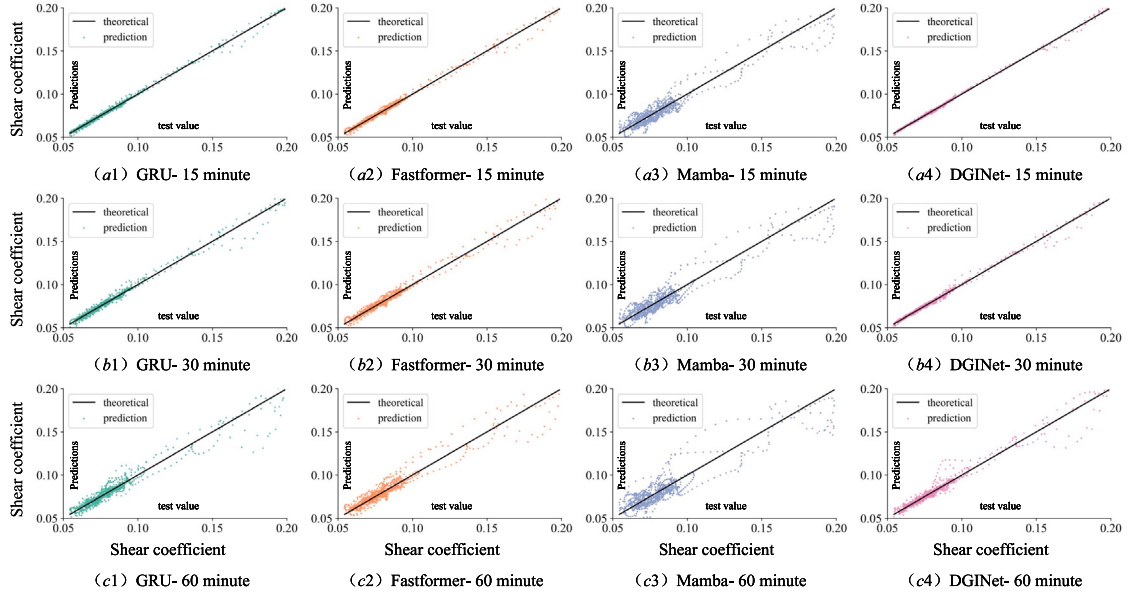


Fig. 12. Scatter plot comparison between prediction track and actual test value of wind shear coefficient.

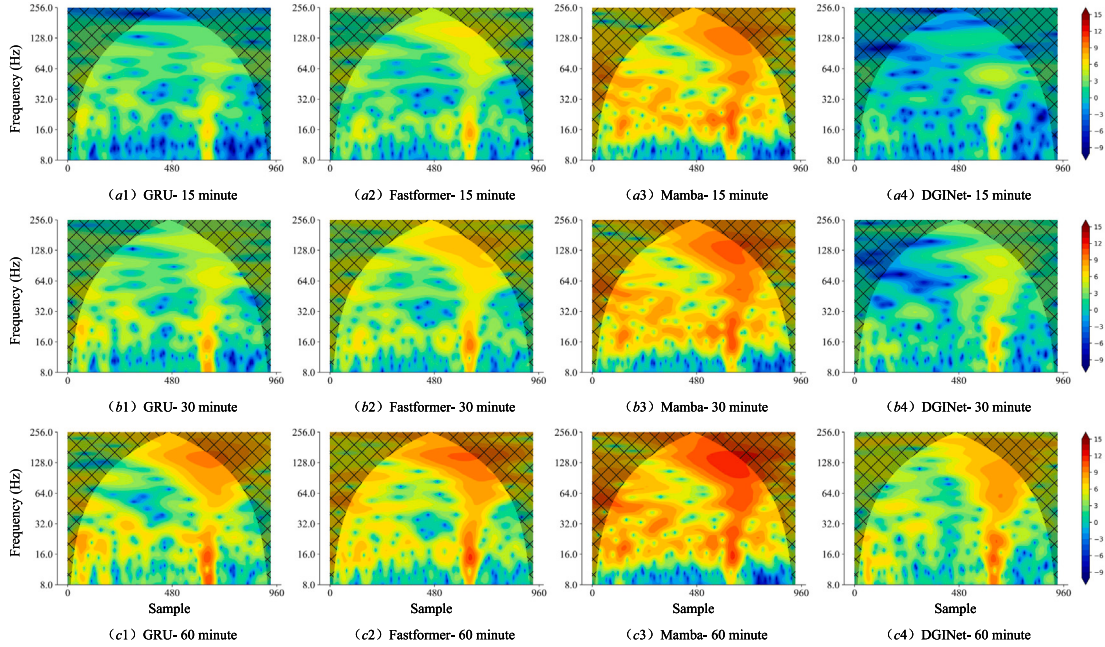


Fig. 13. Comparison of wavelet time-frequency graph of wind shear coefficient prediction errors.

Table 4

Wind speed prediction results for ablation experiments with different time spans (including 15 min, 30 min and 1 h).

Model	Time span								
	15 min			30 min			1 h		
	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2
DGINet	0.051300	0.065637	0.999006	0.091662	0.116207	0.996880	0.170763	0.220009	0.988785
graph2	0.110899	0.140680	0.995435	0.174715	0.220066	0.988810	0.245855	0.315730	0.976904
graph3	0.098474	0.124034	0.996451	0.128719	0.162765	0.993878	0.212346	0.272800	0.982758
graph4	0.099095	0.130965	0.996043	0.141838	0.186426	0.991969	0.227769	0.297832	0.979448
graph5	0.099182	0.125153	0.996387	0.135498	0.173600	0.993036	0.234535	0.298443	0.979364

prediction performance of graph2, graph3, graph4 and graph5 is relatively close, but all lower than the proposed algorithm. However, when predicting for 30 min, the performance of graph2 is significantly worse than the other three models. Subsequent comprehensive comparisons

across all time spans confirmed that graph2 consistently yielded the lowest prediction accuracy. Notably, its performance degradation was most pronounced at the 30-min mark. Compared with the proposed model, the only difference is that the decoder of graph2 is replaced

Table 5

Wind shear coefficient prediction results for ablation experiments with different time spans (including 15 min, 30 min and 1 h).

Model	Time span								
	15 min			30 min			1 h		
	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2
DGINet	0.000837	0.001275	0.997628	0.001309	0.002248	0.992608	0.002933	0.005358	0.957835
graph2	0.001596	0.002112	0.993491	0.002401	0.003167	0.985329	0.002787	0.004739	0.967019
graph3	0.000999	0.001480	0.996803	0.001439	0.002385	0.991680	0.002854	0.005019	0.962996
graph4	0.001196	0.001828	0.995123	0.001638	0.002672	0.989555	0.003034	0.005227	0.959865
graph5	0.001314	0.001825	0.995142	0.001415	0.002435	0.991327	0.003197	0.005274	0.959139

by GRU, which means that the proposed decoder module is sensible. Furthermore, the proposed scheme can be calculated in parallel, while GRU does not support parallel calculation. The observed trend across all models reveals a increase in prediction errors with longer prediction horizons. This because the inherent complexities of long-term wind prediction, linked to the uncertainty and non-stationarity characteristic of wind patterns over extended durations.

Table 5 shows the difference in wind shear prediction performance between the ablation experiments at multiple time spans (including 15 min, 30 min and 1 h). Comprehensively analyzing Tables 5 and 4, due to the difference in magnitude, the prediction error of wind speed still occupies a large weight, while the prediction error of wind shear coefficient is relatively small. As can be seen from Table 5, the prediction results of wind shear coefficient at 15 min and 30 min show that the proposed model still has the lowest MAE and RMSE and the highest R^2 value, which indicates that the proposed model has the highest prediction accuracy. However, the discordant part is that when the time span is extended to one hour, the prediction results of graph2 and graph3 on wind shear coefficient actually exceed the proposed model. However, combined with the wind speed prediction results, the comprehensive prediction results of the proposed model are better. In addition, the encoders of graph2 and graph3 have not changed, which shows that the proposed sample reconstruction method and graph structure data processing method are effective and reasonable. A step back, the decoders of graph2 and graph3 do not support parallel computing, which is the focus of the significant promotion superiority of the proposed model.

6. Conclusion

This paper considers the time-varying characteristics of the wind shear coefficient, realizes the future prediction based on the surface wind speed and the time-varying wind shear coefficient, and can infer the future wind speed at a specified height. It is worth noting that this paper studies the problem of synchronous prediction of low-altitude wind speed and wind shear coefficient, two key variables in the vertical wind profile assessment, at multiple time scales for the first time. This study constructed an ideal multi-task learning model, strengthened the feature expression ability of the backbone network, and simplified the sub-networks for specific tasks. The model adopted an encoder-decoder architecture, proposed a sample reconstruction method within a sliding window, and used improved gated graph unit to construct a graph feature encoder. At the same time, two-dimensional convolution was used to extract the spatial relationship feature encoding of 16-dimensional sample features. The decoder used the output features of the encoder for interactive learning, and then inferred the time series prediction results of wind speed and wind shear coefficient through two neurons respectively. The proposed model shows excellent prediction performance in the time spans of 15 min, 30 min, and 1 h. Using MAE, RMSE, and R^2 as evaluation indicators, the prediction error or deviation from the perfect prediction of the proposed model is significantly reduced, verifying its potential for engineering application.

Overall, this study provides a new method for wind resource assessment, which is expected to improve the accuracy of wind resource and wind power assessment. The limitation of this paper is that if the

evolution pattern of the wind shear coefficient is seriously inconsistent with the training stage, it will lead to large prediction errors. In the future, we will study the calibration and anomaly detection of wind shear coefficient, because the wind shear coefficient varies with seasons and climate, and excessive wind shear will have a negative impact on the normal operation of wind turbines. In addition, we will further explore model explainability [48] to improve reliability in practical applications.

CRedit authorship contribution statement

Ke Fu: Visualization, Investigation, Conceptualization, Writing – original draft, Methodology, Data curation, Software, Formal analysis. **Zhengru Ren:** Resources, Supervision, Validation, Project administration.

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.

Data availability

Data will be made available on request.

References

- [1] G.H.R. Faghani, Z. Najafian Ashrafi, A. Sedaghat, Extrapolating wind data at high altitudes with high precision methods for accurate evaluation of wind power density, case study: Center of Iran, *Energy Convers. Manage.* 157 (2018) 317–338.
- [2] Xinrong Yang, Xin Jiang, Shijing Liang, Yingzuo Qin, Fan Ye, Bin Ye, Jiayu Xu, Xinyue He, Jie Wu, Tianyun Dong, et al., Spatiotemporal variation of power law exponent on the use of wind energy, *Appl. Energy* 356 (2024) 122441.
- [3] Dae-Young Kim, Yeon-Hee Kim, Bum-Suk Kim, Changes in wind turbine power characteristics and annual energy production due to atmospheric stability, turbulence intensity, and wind shear, *Energy* 214 (2021) 119051.
- [4] Yongqian Liu, Yanhui Qiao, Shuang Han, Tao Tao, Jie Yan, Li Li, Galsan Bekhbat, Erdenebat Munkhtuya, Rotor equivalent wind speed calculation method based on equivalent power considering wind shear and tower shadow, *Renew. Energy* 172 (2021) 882–896.
- [5] Can Ma, Taiyu Zhang, Zhiyu Jiang, Zhengru Ren, Dynamic analysis of lowering operations during floating offshore wind turbine assembly mating, *Renew. Energy* (2025) 122528.
- [6] P. Valsaraj, Drisya Alex Thumba, K. Asokan, K. Satheesh Kumar, Symbolic regression-based improved method for wind speed extrapolation from lower to higher altitudes for wind energy applications, *Appl. Energy* 260 (2020) 114270.
- [7] Valerij Bezrukovs, Aleksejs Zacepins, Vladislavs Bezrukovs, Vitalijs Komasilovs, Comparison of methods for evaluation of wind turbine power production by the results of wind shear measurements on the Baltic shore of Latvia, *Renew. Energy* 96 (2016) 765–774.
- [8] Christopher Jung, Dirk Schindler, Development of a statistical bivariate wind speed-wind shear model (WSWS) to quantify the height-dependent wind resource, *Energy Convers. Manage.* 149 (2017) 303–317.
- [9] Giovanni Gualtieri, Atmospheric stability varying wind shear coefficients to improve wind resource extrapolation: A temporal analysis, *Renew. Energy* 87 (2016) 376–390.
- [10] Giovanni Gualtieri, Surface turbulence intensity as a predictor of extrapolated wind resource to the turbine hub height: method's test at a mountain site, *Renew. Energy* 120 (2018) 457–467.

- [11] Giovanni Gualtieri, Sauro Secci, Methods to extrapolate wind resource to the turbine hub height based on power law: A 1-h wind speed vs. Weibull distribution extrapolation comparison, *Renew. Energy* 43 (2012) 183–200.
- [12] Giovanni Gualtieri, A comprehensive review on wind resource extrapolation models applied in wind energy, *Renew. Sustain. Energy Rev.* 102 (2019) 215–233.
- [13] Paola Crippa, Mariana Alifa, Diogo Bolster, Marc G. Genton, Stefano Castruccio, A temporal model for vertical extrapolation of wind speed and wind energy assessment, *Appl. Energy* 301 (2021) 117378.
- [14] Ebubekir Firtun, Önder Güler, Seyit Ahmet Akdağ, Investigation of wind shear coefficients and their effect on electrical energy generation, *Appl. Energy* 88 (11) (2011) 4097–4105.
- [15] Md Saiful Islam, Mohamed Mohandes, Shafiqur Rehman, Vertical extrapolation of wind speed using artificial neural network hybrid system, *Neural Comput. Appl.* 28 (2017) 2351–2361.
- [16] Simon H. Lee, Paul D. Williams, Thomas H.A. Frame, Increased shear in the north atlantic upper-level jet stream over the past four decades, *Nature* 572 (7771) (2019) 639–642.
- [17] F.-F. Jin, J. Boucharel, I.-I. Lin, Eastern Pacific tropical cyclones intensified by El Niño delivery of subsurface ocean heat, *Nature* 516 (7529) (2014) 82–85.
- [18] Jiale Li, Xuefei Wang, Xiong Bill Yu, Use of spatio-temporal calibrated wind shear model to improve accuracy of wind resource assessment, *Appl. Energy* 213 (2018) 469–485.
- [19] Yaganteeswarudu Akkem, Saroj Kumar Biswas, Aruna Varanasi, Smart farming using artificial intelligence: A review, *Eng. Appl. Artif. Intell.* 120 (2023) 105899.
- [20] Junqiang Wei, Xuejie Wu, Tianming Yang, Runhai Jiao, Ultra-short-term forecasting of wind power based on multi-task learning and LSTM, *Int. J. Electr. Power Energy Syst.* 149 (2023) 109073.
- [21] Qishu Liao, Di Cao, Zhe Chen, Frede Blaabjerg, Weihao Hu, Probabilistic wind power forecasting for newly-built wind farms based on multi-task Gaussian process method, *Renew. Energy* 217 (2023) 119054.
- [22] Yugui Tang, Kuo Yang, Shujing Zhang, Zhen Zhang, Wind power forecasting: A hybrid forecasting model and multi-task learning-based framework, *Energy* 278 (2023) 127864.
- [23] Saptadeep Biswas, Gyan Singh, Binanda Maiti, Absalom El-Shamir Ezugwu, Kashif Saleem, Aseel Smerat, Laith Abualigah, Uttam Kumar Bera, Integrating differential evolution into gazelle optimization for advanced global optimization and engineering applications, *Comput. Methods Appl. Mech. Engrg.* 434 (2025) 117588.
- [24] Sunilkumar P. Agrawal, Pradeep Jangir, Laith Abualigah, Sundaram B. Pandya, Anil Parmar, Absalom E. Ezugwu, Aseel Smerat, et al., The quick crisscross sine cosine algorithm for optimal FACTS placement in uncertain wind integrated scenario based power systems, *Results Eng.* 25 (2025) 103703.
- [25] Yaganteeswarudu Akkem, Saroj Kumar Biswas, Aruna Varanasi, A comprehensive review of synthetic data generation in smart farming by using variational autoencoder and generative adversarial network, *Eng. Appl. Artif. Intell.* 131 (2024) 107881.
- [26] Jinhao Shi, Bo Wang, Kaiyi Luo, Yifei Wu, Min Zhou, Junzo Watada, Ultra-short-term wind power interval prediction based on multi-task learning and generative critic networks, *Energy* 272 (2023) 127116.
- [27] Lei Wang, Yigang He, M2STAN: Multi-modal multi-task spatiotemporal attention network for multi-location ultra-short-term wind power multi-step predictions, *Appl. Energy* 324 (2022) 119672.
- [28] Manuel Dorado-Moreno, Nicolò Navarin, Pedro Antonio Gutiérrez, Luis Prieto, Alessandro Sperduti, Sancho Salcedo-Sanz, César Hervás-Martínez, Multi-task learning for the prediction of wind power ramp events with deep neural networks, *Neural Netw.* 123 (2020) 401–411.
- [29] Tian-Bao Li, An-An Liu, Dan Song, Wen-Hui Li, Jing Zhang, Zhi-Qiang Wei, Yu-Ting Su, Multi-task spatial-temporal transformer for multi-variable meteorological forecasting, *IEEE Trans. Knowl. Data Eng.* (2024).
- [30] Yuejiang Chen, Jiang-Wen Xiao, Yan-Wu Wang, Yuanzheng Li, Regional wind-photovoltaic combined power generation forecasting based on a novel multi-task learning framework and TPA-LSTM, *Energy Convers. Manage.* 297 (2023) 117715.
- [31] Yu Zheng, Huan Yee Koh, Ming Jin, Lianhua Chi, Haishuai Wang, Khoa T Phan, Yi-Ping Phoebe Chen, Shirui Pan, Wei Xiang, Graph spatiotemporal process for multivariate time series anomaly detection with missing values, *Inf. Fusion* 106 (2024) 102255.
- [32] Lijun Sun, Mingzhi Liu, Guanfeng Liu, Xiao Chen, Xu Yu, FD-TGCN: Fast and dynamic temporal graph convolution network for traffic flow prediction, *Inf. Fusion* 106 (2024) 102291.
- [33] Xin Liu, Jingjia Yu, Lin Gong, Minxia Liu, Xi Xiang, A GCN-based adaptive generative adversarial network model for short-term wind speed scenario prediction, *Energy* 294 (2024) 130931.
- [34] Wei-Chia Huang, Chiao-Ting Chen, Chi Lee, Fan-Hsuan Kuo, Szu-Hao Huang, Attentive gated graph sequence neural network-based time-series information fusion for financial trading, *Inf. Fusion* 91 (2023) 261–276.
- [35] ZhuoLin Li, ZiHeng Gao, GaoWei Zhang, JingJing Liu, LingYu Xu, Dynamic personalized graph neural network with linear complexity for multivariate time series forecasting, *Eng. Appl. Artif. Intell.* 127 (2024) 107291.
- [36] Jelena Simeunović, Baptiste Schubnel, Pierre-Jean Alet, Rafael E. Carrillo, Spatio-temporal graph neural networks for multi-site PV power forecasting, *IEEE Trans. Sustain. Energy* 13 (2) (2021) 1210–1220.
- [37] Ziheng Gao, Zhuolin Li, Lingyu Xu, Jie Yu, Dynamic adaptive spatio-temporal graph neural network for multi-node offshore wind speed forecasting, *Appl. Soft Comput.* 141 (2023) 110294.
- [38] Ziheng Gao, Zhuolin Li, Haoran Zhang, Jie Yu, Lingyu Xu, Dynamic spatiotemporal interactive graph neural network for multivariate time series forecasting, *Knowl.-Based Syst.* 280 (2023) 110995.
- [39] Yuyan Wang, Jie Hu, Fei Teng, Lilan Peng, Shengdong Du, Tianrui Li, KaTaGCN: Knowledge-augmented and time-aware graph convolutional network for efficient traffic forecasting, *Inf. Fusion* 111 (2024) 102542.
- [40] Weikai Ren, Ningde Jin, Lei OuYang, Phase space graph convolutional network for chaotic time series learning, *IEEE Trans. Ind. Inform.* (2024).
- [41] Thomas N. Kipf, Max Welling, Semi-supervised classification with graph convolutional networks, 2016, arXiv preprint arXiv:1609.02907.
- [42] Minhao Liu, Ailing Zeng, Muxi Chen, Zhijian Xu, Qiuxia Lai, Lingna Ma, Qiang Xu, Scinet: Time series modeling and forecasting with sample convolution and interaction, *Adv. Neural Inf. Process. Syst.* 35 (2022) 5816–5828.
- [43] Chen Liu, Juntao Zhen, Wei Shan, Time series classification based on convolutional network with a gated linear units kernel, *Eng. Appl. Artif. Intell.* 123 (2023) 106296.
- [44] Noam Shazeer, Glu variants improve transformer, 2020, arXiv preprint arXiv:2002.05202.
- [45] Ahmad MohdAziz Hussein, Saleh Ali Alomari, Mohammad H Almomani, Raed Abu Zitar, Kashif Saleem, Aseel Smerat, Shawd Nusier, Laith Abualigah, A smart IoT-cloud framework with adaptive deep learning for real-time epileptic seizure detection, *Circuits Systems Signal Process.* 44 (3) (2025) 2113–2144.
- [46] Y. Akkem, B.S. Kumar, A. Varanasi, Streamlit application for advanced ensemble learning methods in crop recommendation systems—a review and implementation, *Indian J. Sci. Technol.* 16 (48) (2023) 4688–4702.
- [47] Yaganteeswarudu Akkem, Saroj Kumar Biswas, Analysis of an intellectual mechanism of a novel crop recommendation system using improved heuristic algorithm-based attention and cascaded deep learning network, *IEEE Trans. Artif. Intell.* (2024).
- [48] Yaganteeswarudu Akkem, Saroj Kumar Biswas, Aruna Varanasi, Streamlit-based enhancing crop recommendation systems with advanced explainable artificial intelligence for smart farming, *Neural Comput. Appl.* 36 (32) (2024) 20011–20025.