DSformer: Integration of DeepResGCN and SMARTformer for Enhanced Spatiotemporal Wind Speed Forecasting

Chenxing Zhu[1],Yizhi Dong[2],Zijun Ye[3] and Wenbo Wang[4*]

***Abstract*—**

It is crucial for wind power forecasting to predict wind speed with high accuracy, as this directly impacts the performance of the entire power system. However, existing wind speed forecasting methods have failed to fully exploit the correlations among multiple neighboring wind farms. Therefore, in light of the necessity to comprehensively account for these interdependencies, we introduce a novel wind speed forecasting model, DSformer, which integrates a Deep Residual Graph Convolutional Network (DeepResGCN) based on deep residual stacking with a Semi-Autoregressive Transformer (SMARTformer) that incorporates an integrated window attention mechanism.Specifically, DeepResGCN is employed for extracting the spatial features of wind speed, while SMARTformer is utilized to learn its temporal characteristics. Furthermore, we characterize the correlations from both the intensity and time-lag dimensions and adopt a multi-graph feature fusion method based on joint low-rank decomposition and inverse reconstruction to integrate the features from these two dimensions. As a result, DSformer is able to more comprehensively capture the spatio-temporal correlations among adjacent wind farms.Moreover, in a case study involving data from 15 wind farms, DSformer achieved average absolute errors (MAE) of 0.147m/s, 0.189m/s, and 0.238m/s, and root mean square errors (RMSE) of 0.273m/s, 0.291m/s, and 0.357m/s for 4-, 6-, and 8-hour forecasts respectively. In comparison with existing methods, DSformer demonstrates superior prediction accuracy and efficiency, ultimately making it a promising solution for the challenges encountered in wind power forecasting.

I. INTRODUCTION

As the global energy transition accelerates toward low-carbon systems, wind energy has established itself as a linchpin in carbon neutrality roadmaps, leveraging its dual advantages of technological readiness and scalable deployment. Its widespread adoption is pivotal for energy system transformation, particularly manifested in replacing fossil fuels for carbon reduction, optimizing energy security frameworks, and driving paradigmatic shifts in energy economics. Within China's renewable energy transition framework, accurate wind speed forecast is of great engineering value in enhancing wind power integration efficiency, ensuring stable power system operation, and optimizing multi-energy complementarity scheduling strategies.

However, the inherent intermittency and stochasticity of wind energy drive spatiotemporal variations in wind speed, consequently inducing substantial fluctuations in power generation spanning minute-level to seasonal timescales. High-precision wind speed prediction holds critical potential to mitigate grid disturbances induced by wind power volatility, enhance operational stability of wind generation systems, and improve grid efficiency. Such predictive capabilities prove indispensable for optimizing grid dispatch strategies, reducing wind curtailment rates, elevating economic returns from wind energy, and ensuring secure power system operations. Consequently, the intrinsic instability of wind speed variations poses formidable challenges for achieving timely and reliable wind speed forecasting essential for modern energy systems[2].

Current approaches to wind speed prediction can be divided into five principal frameworks: physics-driven models, statistical models, machine learning approaches, deep learning frameworks, and hybrid models. Physics-driven methods rely on atmospheric dynamics equations coupled with Numerical Weather Prediction (NWP) [11] for spatiotemporal extrapolation. Statistical models, such as Autoregressive Integrated Moving Average (ARIMA) and Kalman filtering, establish linear and stationary predictions through temporal feature mining. Machine learning methods, including Support Vector Machines (SVM) [13] and Random Forests (RF) [16], construct nonlinear mappings between wind speed and correlated variables via feature engineering. Deep learning frameworks leverage neural networks to autonomously extract dynamic patterns from wind speed sequences, exemplified by Convolutional Neural Networks (CNN) [17], Long Short-Term Memory (LSTM) networks, and Transformers [10]. Hybrid models integrate diverse methodologies to enhance robustness. For in-

[1]Chenxing Zhu is an undergraduate student majoring in Statistics at Wuhan University of Science and Technology, Wuhan, China. 13667121392@163.com

[2]Yizhi Dong is an undergraduate student majoring in Statistics at Wuhan University of Science and Technology, Wuhan, China. sueiran42526@163.com

[3]Zijun Ye is an undergraduate student majoring in Statistics at Wuhan University of Science and Technology, Wuhan, China. yezijunchongyaaa@163.com

[4*]Wenbo Wang is a professor at Wuhan University of Science and Technology, Wuhan, China. wangwenbo@wust.edu.cn

stance, Wang et al [14]. validated the effectiveness of decomposition-ensemble strategies (e.g., empirical mode decomposition combined with LSTM) for non-stationary wind speed prediction. Literature [12] proposed a cluster-based forecasting model using K-means clustering to partition wind speed fluctuation patterns, integrated with LSTM and ARIMA.

Currently, spatiotemporal fusion models have emerged as a pivotal research frontier in prediction accuracy enhancement through integrated spatiotemporal dynamics modeling.In joint spatiotemporal modeling, Fan et al [9] introduced a Spatiotemporal Neural Network (ST-NN), employing CNNs to extract spatial correlations among adjacent meteorological stations and Bidirectional Gated Recurrent Units (BiGRU) to model long-term temporal dependencies. Building on this, literature [15] expanded input features to multidimensional data (e.g., wind speed, temperature, and pressure) and designed a CNN-LSTM hybrid architecture for multifactor spatiotemporal modeling. Wang et al.[7] experimentally demonstrated that spatial correlation mining significantly enhances prediction performance, providing theoretical foundations for subsequent studies. In spatial correlation modeling, researchers have transcended traditional grid-based approaches by innovatively applying Graph Neural Networks (GNNs) to construct topological relationships among wind farm nodes. Literature [1] emphasized that GNNs effectively model topological interactions between wind farms, offering novel perspectives for spatial correlation analysis.

Transformer models exhibit unique advantages in time series forecasting, primarily due to their self-attention mechanisms for capturing long-range dependencies. However, conventional Transformers suffer from critical limitations in temporal tasks: channel-mixing encoders distort feature spaces, attention matrices incur high computational complexity, and static weights struggle to adapt to dynamic spatiotemporal correlations [5]. To address these issues, researchers have proposed improved variants such as Informer [8], PatchTST [4], and Crossformer [6]. Nevertheless, challenges persist, such as excessive coupling of channel interactions and insufficient modeling of dynamic spatial relationships. We adopt the SMARTformer [3] model for wind speed prediction, which simultaneously captures local and global dependencies by combining autoregressive and non-autoregressive advantages, thereby avoiding feature entanglement across periods and significantly enhancing time series forecasting performance.

We focus on short-term spatiotemporal wind speed prediction across multiple wind farms and propose DSformer, a novel forecasting model based on enhanced DeepResGCN and SMARTformer. Specifically, our framework consists of three interconnected stages:

- **Dynamic Graph Construction**: We construct dynamic graphs using geographical adjacency and historical correlation matrices.
- **Hierarchical Spatial Feature Extraction**: We extract hierarchical spatial features via DeepResGCN.
- **Multi-granularity Temporal Dependency Modeling**: We model multi-granularity temporal dependencies through SMARTformer's segmented window attention mechanism.

Through rigorous comparative experiments on a 15-wind-farm dataset spanning three climatic zones, we demonstrate DSformer's consistent superiority over seven state-of-the-art baselines in 4~8-hour forecasting horizons, thereby validating its methodological efficacy and operational feasibility. Furthermore, this work establishes a robust computational foundation for optimizing wind farm dispatch strategies and enhancing grid-connected renewable energy management systems.

II. METHODOLOGY

A. Spatial Feature Extraction based on DeepResGCN

1) Graph Construction: We construct a graph $G = (V, E)$ with wind farms as nodes and their adjacency as edges. First we construct spatial and temporal adjacency matrices. The spatial matrix includes distance and direction, while the temporal one uses wind speed correlations. These matrices capture both static and dynamic wind field interactions.

We integrate geographical distance d_{dist}, wind direction angular distance d_{angel}, temperature distance d_{temp}, and pressure distance d_{press} between wind farms into a composite spatial distance metric $d = (d_{ij}) \in \mathbb{R}^{N\times N}$, where N is the total number of wind farms. The integration is formulated as:

$$d_{ij} = d_{ij}^{\text{dist}} + d_{ij}^{\text{angel}} + d_{ij}^{\text{temp}} + d_{ij}^{\text{press}} \tag{1}$$

where d_{ij} represents the distance metric between the i-th and j-th wind farms.

Then we apply Gaussian kernel normalization to obtain the spatial adjacency matrix, where σ represents the bandwidth parameter:

$$\Lambda_S(i,j) = \exp\left(-\frac{d_{ij}^2}{2\sigma^2}\right) \tag{2}$$

To characterize temporal delays across wind farms, we compute Pearson correlation coefficients between wind speed time series pairs (i, j) at multiple time lags. Coefficients below a predefined threshold are discarded, subsequently generating the time-lagged adjacency matrix.

$$\rho_{ij} = \frac{\sum(x_i - \bar{x}_i)(x_j - \bar{x}_j)}{\sqrt{\sum(x_i - \bar{x}_i)^2}\sqrt{\sum(x_j - \bar{x}_j)^2}} \tag{3}$$

where the wind speed sequence of the i-th wind farm is denoted as x_i.

2) Adjacency Matrix Fusion: We propose a multi-graph feature fusion method based on joint low-rank decomposition and inverse reconstruction to integrate static and dynamic features, avoiding the information loss of conventional weighted fusion methods and reducing complexity via latent factor constraints.

First we decompose temporal and spatial adjacency matrices into a shared latent space via non-negative matrix factorization (NMF), obtaining a common basis matrix $W \in \mathbb{R}^{N\times k}$ and view-specific coefficient matrices $H_T, H_S \in \mathbb{R}^{k\times N}$:

$$A_{\text{Temporal}} \approx WH_T, \quad A_{\text{Spatial}} \approx WH_S \tag{4}$$

where $k \ll N$ denotes the latent factor dimension, W encodes spatiotemporal common features.

Then we perform feature-level fusion on the coefficient matrices H_T and H_S by employing the Hadamard product:

$$H_{\text{fused}} = H_T \odot H_S \tag{5}$$

At last, we reconstruct the output matrix $A_{\text{fused}} \in \mathbb{R}^{N\times N}$ via the basis matrix W and the fused coefficient matrix H_{fused} ,formulated as:

$$A_{\text{fused}} = W H_{\text{fused}} \tag{6}$$

3) Multi-layer Stacked Graph Convolutional Networks: We use multi-layer stacking to integrate long-range neighborhood info. Each GCN layer captures features from interactions as the network deepens, enhancing discriminative power. We introduce residual connections. The layer output depends on both the current transformation and original input, stabilizing gradient propagation and preserving features. The residual connection is formulated as:

$$\begin{aligned} H^{(l+1)} =& \text{ReLU}\left(\text{LayerNorm}\left(\text{GCN}\left(H^{(l)}, A_{\text{fused}}\right)\right.\right. \\ &\left.\left. + H^{(l)}\right)\right) \end{aligned} \tag{7}$$

where $H^{(l)} \in \mathbb{R}^{(N\cdot N)\times d}$ denotes the input at layer l, with N being the number of nodes and d the feature dimension. The operation $\text{GCN}(H^{(l)}, A_{\text{fused}})$ indicates the graph convolution output from $H^{(l)}$ using the fused spatiotemporal adjacency matrix A_{fused}. The LayerNorm$(\cdot)$ stabilizes outputs across layers, while ReLU$(\cdot)$ serves as the activation function. The GCN process is formulated as follows:

$$\text{GCN}\left(H^{(l)}, A_{\text{fused}}\right) = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}_{\text{fused}}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right) \tag{8}$$

where $H^{(l)} \in \mathbb{R}^{N\times d}$ is the input feature matrix for layer l, $\widetilde{A}_{\text{fused}} = A_{\text{fused}} + I$ is the self-loop-augmented adjacency matrix, $\widetilde{D}$ is the degree matrix of $\widetilde{A}_{\text{fused}}$, $W^{(l)}$ is the learnable weight matrix for the l-th layer, and σ denotes the activation function.

The model captures higher-order neighborhood information by stacking L residual blocks, ultimately generating deep feature representations $H^{(L)} \in \mathbb{R}^{(N\cdot N)\times d}$.

Then we perform feature fusion via element-wise weighted pooling. As residual block depth grows, aggregated info expands and inter-layer aggregation approaches convergence. We use an exponentially decaying coefficient $\lambda(0 < \lambda < 1)$ and assign weight λ^{k-i} to the i-th matrix (k is the matrix count). Larger indices i receive greater weights through this scheme. The normalized weights are then formulated as:

$$w_i = \frac{\lambda^{k-i}}{\sum_{j=1}^{k}\lambda^{k-j}} \tag{9}$$

For $L{+}1$ layer outputs, where $l = 0$ represents the original input $X^{(0)}$, we perform element-wise weighted averaging on each channel for every layer output $X^{(l)}$. This operation yields the fused feature matrix Y:

$$Y = \frac{1}{L+1}\sum_{l=1}^{L} w_l X^{(l)} \tag{10}$$

The framework of DeepResGCN is illustrated in Figure 1.

B. A Temporal Feature Extraction Model Based on SMARTformer

1) Time-Independent Embedding: For each timestep, the embedding process is conducted in two components:

1) **Value Embedding**: We perform embedding of raw data at each timestep into a C_v-dimensional vector via 1D convolution;
2) **Time-Independent Embedding** F_i: This is computed as the concatenation of three temporal features:

$$F_i = \left(E_i^{(m/h)} + E_i^{(wk)} + E_i^{(mth)}\right) \tag{11}$$

where $E_i^{(m/h)}, E_i^{(wk)}, E_i^{(mth)}$ denote three learnable projection matrices for positional embeddings of minute/hour, weekday and month, respectively. This hierarchical sharing mechanism enables positional information to propagate across timesteps of identical periodic patterns (e.g., daily, weekly, or monthly), thereby strengthening the model's capacity to discern global temporal dependencies.

For each token $\chi_{en}^{1:T} = \{Y_{i,d} | 1 \le i \le T, 1 \le d \le D\}$, where $Y_{i,d}$ denotes the output of the DeepResGCN model at the i-th timestep in the d-th dimension, we concatenate and normalize the two embeddings for every Y_i:

$$E_i = \text{norm}(\text{Conv1D}(Y_i) \parallel F_i) \tag{12}$$

where $\text{Conv1D}(Y_i) \in \mathbb{R}^{C_v}$ represents the latent variable after value embedding, and $E_i \in \mathbb{R}^C$ (with $C = C_v + C_p$) denotes the input embedding. This process leverages positional information without distorting data semantics, thereby capturing global features of ultra-long-term temporal variations.

2) Integrated Window Attention: The Integrated Window Attention mechanism projects input features X into K heads, divided into S for intra-window and $K - S$ for inter-window attention. This reduces computation and parameters, and adaptively adjusts attention regions for efficient data variation capture.

For attention in windows, X is segmented into non-overlapping temporal windows of length w, where w is a divisor of the total sequence length T. The window size w is tunable to balance model expressiveness and computational demands. Based on this partitioning, intra-window attention can be defined as:

$$\begin{aligned} &X = \left[X^1, X^2, \ldots, X^M\right] \\ &Y_k^i = \text{Attention}\left(X^i W_k^Q, X^i W_k^K, X^i W_k^V\right) \\ &\text{Intra-Attention}(X) = \left[Y_k^1, Y_k^2, \ldots, Y_k^M\right] \end{aligned} \tag{13}$$

where $X^i \in \mathbb{R}^{\frac{T}{M}\times C}$ and $M = T/w$ $(i = 1, 2, \ldots, M)$. $W_k^Q \in \mathbb{R}^{C\times d_k}, W_k^K \in \mathbb{R}^{C\times d_k}$ and $W_k^V \in$

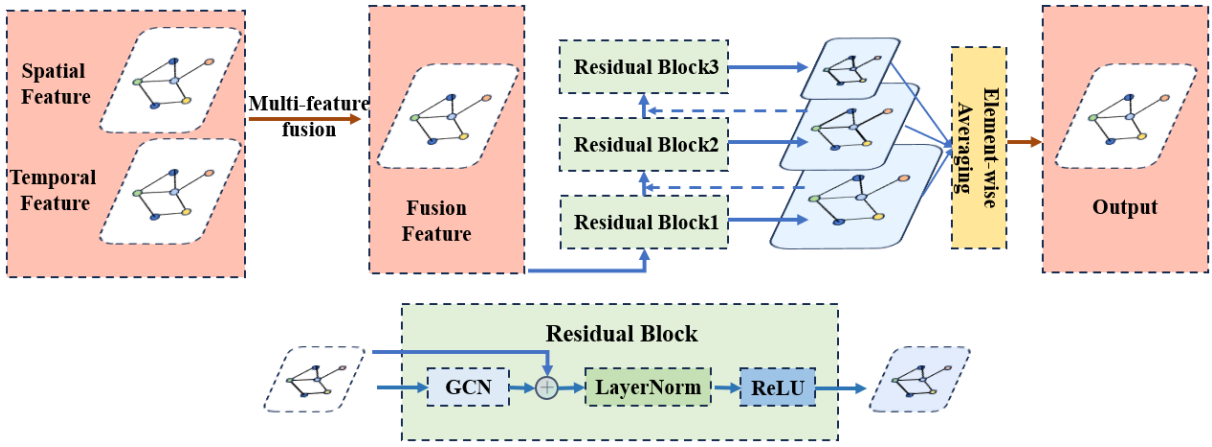

Fig. 1: Framework diagram of the Spatial Feature Extraction based on DeepResGCN

$\mathbb{R}^{C\times d_k}$ represent the projection matrices of queries, keys, and values for the k-th head respectively where d_k is set as C/K.

For the remaining $K-S$ heads, we perform Inter-Attention by shifting the original sequence X by os timesteps to generate X_s, which is then partitioned into non-overlapping windows of size w (identical to the previous branch) to obtain $\hat{X}$:

$$\begin{aligned} X_s &= (X[os:T, 0:C] \| X[0:os, 0:C]) \\ \hat{X} &= [X_s^1, X_s^2, \ldots, X_s^M] \end{aligned} \tag{14}$$

The shifted sequence $\hat{X}$ is projected as Queries and X is projected as Keys and Values. Then, we acquire the k-th head Y_k^i for the i-th window as attention scores.

$$\begin{aligned} Y_k^i &= \text{Attention}\left(\hat{X}^i W_k^K, X^i W_k^K, X^i W_k^V\right) \\ \text{Inter-Attention}(X) &= \left[Y_k^1, Y_k^2, \ldots, Y_k^M\right] \end{aligned} \tag{15}$$

where $ew < os < (e+1)w, 0 < e < M$. This mechanism facilitates robust inter-window connections and enhances interactions among edge tokens through optimized attention allocation. Then, we integrate two branches to acquire the final output as

$$\begin{aligned} &\text{IntWin-Attention}(X) = (Y_1 \| \ldots \| Y_k) \\ &Y_k = \begin{cases} \text{Intra-Attention}_k(X), & \text{if } 1 \le k \le S \\ \text{Inter-Attention}_k(X), & \text{if } S+1 \le k \le K \end{cases} \end{aligned} \tag{16}$$

3) Semi-Autoregressive Decoder: We present a hierarchical semi-autoregressive (SAR) decoder architecture, which integrates a segment autoregressive (AR) layer and a non-autoregressive (NAR) refining layer. Both layers share the structural framework of conventional Transformer decoders but replace the standard attention mechanism with the Integrated Window Attention (IWA).

We introduce a segment-wise autoregressive decoder layer at the network's lower level, which iteratively applies identical decoder layers to predict subsequences. For a long sequence of length L, the model partitions it into k non-overlapping segments, each of length l_k, during prediction at the M-th decoder layer. The output of the M-th decoder layer $Z_{de}^M \in \mathbb{R}^{L\times C}$ is obtained by concatenating representations from all k steps along the temporal dimension, as follows:

$$Y_{de}^M = \left(\hat{Y}_{de}^M(1) \| \ldots \| \hat{Y}_{de}^M(k)\right) \tag{17}$$

where $\hat{Y}_{de}^M(j)$ denotes the j-th step output. This process enables the propagation of feature information within local wind speed sequences.

In the j-th step, we assume L_j for the predicted length. A local representation $H_{(j)}^M \in \mathbb{R}^{L_j\times C}$ takes as the input, which comes from prior predictions and current positional info to capture local wind speed variations. $\hat{Y}_{de}^M(j-1) \in \mathbb{R}^{L_{j-1}\times C}$ represents the slices extracted from the prior forecast. After temporal alignment via padding or cropping, positional embeddings $F_{(j)} \in \mathbb{R}^{L_j\times C_p}$ using temporally independent embeddings (TIE) to enhance the modeling of complex periodic changes. These embeddings are concatenated and learned through a MLP, as shown below:

$$H_{(j)}^M = \text{MLP}\left(F_{(j)} \| \hat{Y}_{de}^M(j-1)\right) \tag{18}$$

The equation to perform in the Segment AR decoder can be summarized as

$$\hat{Y}_{de}^M(j) = \text{Decoder}\left(H_{(j)}^M, Y_{en}^N\right) \tag{19}$$

where Y_{en}^N represents the final output after all N layers of the encoder. The predicted length for each step can be determined by the dataset's sampling frequency, and selecting a length that is an integer multiple of the dataset period (day/week) can aid model prediction.

We enhance global representational capacity beyond segmented autoregressive layers by incorporating a NAR Refining layer, where the $(M+1)$th refinement layer takes input from the preceding layer's hidden states $Y_{de}^M \in \mathbb{R}^{L\times C}$. And the equations of the $(M+1)$th decoder layer can be summarized as:

$$Y_{de}^{M+1} = Decoder\left(Y_{de}^M, Y_{en}^N\right) \tag{20}$$

We alternate stacking the two layers to capture global and local contexts. This design combines NAR's global horizon with AR's local detail-capturing ability, enhancing the decoder's capacity for dependencies.

C. DSformer time series forecasting model

Figure 2 shows the DSformer framework, where residual blocks integrate GCN modules and SMARTformer components. The DeepResGCN network extracts detailed spatial wind speed features, while the SMARTformer module's temporal-independent embeddings, windowed attention, and semi-autoregressive decoders enhance long-term temporal modeling. This holistic spatiotemporal combination achieves robust short-term wind speed forecasting, with experiments confirming superior performance and reliability.

III. RESULTS AND DISCUSSION

A. Description of the dataset

We utilize data from the Inner Mongolia region of the Asia-Pacific Sunflower Solar Dataset within the Wind Integration National Dataset (WIND) Toolkit for wind speed prediction. The dataset includes wind speed, wind direction, temperature, and air pressure from 15 wind farms in 2020. It has a 10-minute temporal resolution and spatial distances of 7–70 km. The total dataset consists of $15 \times 6 \times 24 \times 365 = 15 \times 52{,}560$ data frames. The data are divided into training (70%), validation (10%), and test sets (20%) based on time order.

B. Evaluation Metrics AND Hyperparameter Tuning

We employ the Mean Squared Error (MSE) and Mean Absolute Error (MAE) to comprehensively evaluate the spatiotemporal wind speed prediction performance of the DSformer model.To comprehensively validate the effectiveness of the proposed model, we conduct comparative experiments against five representative benchmark models spanning both traditional machine learning and deep learning paradigms.The hyperparameter settings are shown in Table I.

TABLE I: Hyperparameter Tuning for Benchmark Models and DSformer

Model	**Hyperparameter Tuning**
Informer	Based on publicly available hyperparameter tuning (learning rate, number of attention heads, encoder layers, decoder layers, embedding dimension, dropout rate).
LSTM	Grid search (hidden layer size, learning rate, dropout rate, sequence length, batch size).
SMARTformer	Based on publicly available hyperparameter tuning (similar to Informer).
PatchTST	Based on publicly available hyperparameter tuning (similar to Informer).
LightGBM	Grid search (hyperparameters including learning rate, number of iterations, maximum tree depth, number of leaf nodes, feature subsampling ratio, and regularization parameters).
DSformer	Learning rate, graph convolutional layers, SMARTformer layers, embedding dimension, dropout rate, adjacency matrix construction method.

TABLE II: Comparison of MSE for Prediction Results of Different Algorithms

Model	**Advanced Time**		
	4 h	**6 h**	**8 h**
PatchTST	0.155	0.205	0.265
Informer	0.178	0.224	0.288
LSTM	0.210	0.296	0.374
SMARTformer	0.192	0.286	0.356
LightGBM	0.241	0.242	0.368
DSformer	**0.147**	**0.189**	**0.238**

C. Main Results

We compare DSformer with other baseline models, with results summarized in Tables II and III.

This result strongly validates the effectiveness of the DSformer model. DSformer can better learn the spatiotemporal dependencies of wind speed data by deeply modeling spatial relationships through DeepResGCN and effectively capturing temporal features through SMARTformer. We elaborate from three aspects:

1) **Short-term Prediction Performance:** In the 4-hour prediction task, DSformer achieves an MSE of 0.147 and an MAE of 0.273, representing reductions of 5.2% and 4.2%, respectively, compared to the second-best model, PatchTST.
2) **Medium- to Long-term Prediction Advantages:** As the prediction time span increases, DSformer's performance advantage becomes more pronounced. In the 6-hour prediction, it achieves an MSE of 0.189 and an MAE of 0.291, representing reductions of 7.8% and 4.5%, respectively, compared to PatchTST.
3) **Long-term Prediction Robustness:** In the 8-hour prediction task, DSformer achieves an MSE of 0.238 and an MAE of 0.357, representing reductions of 11.3% and 5%, respectively, compared to the optimal baseline model, PatchTST. This performance is significantly better than that of traditional methods (for example, LightGBM's MSE increases by 52.7%), verifying the model's effectiveness in suppressing error accumulation.

In Figure 3, we compare DSformer's 6-hour wind speed prediction results with those of other models. It integrates DeepResGCN's spatial feature extraction and SMARTformer's temporal dependency modeling, leveraging their strengths for handling nonlinear and non-stationary wind speed data. This combination gives DSformer a competitive edge in long-term forecasting.

D. Ablation Study and Analysis

Ablation studies on the DeepResGCN and SMARTformer modules of the DSformer model con-

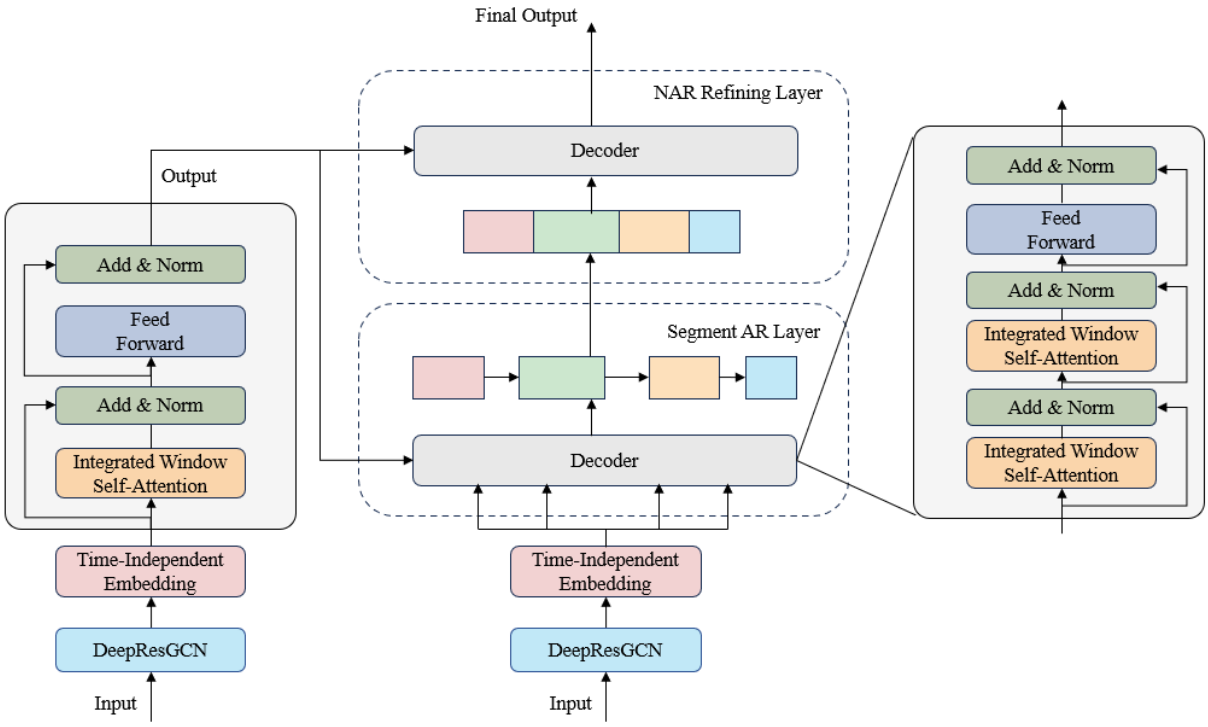

Fig. 2: Architectural diagram of the DSformer model

TABLE III: Comparison of MAE for Prediction Results of Different Algorithms

Model	Advanced Time		
	4 h	6 h	8 h
PatchTST	0.285	0.312	0.375
Informer	0.316	0.345	0.402
LSTM	0.355	0.416	0.488
SMARTformer	0.331	0.407	0.445
LightGBM	0.412	0.431	0.539
DSformer	**0.273**	**0.291**	**0.357**

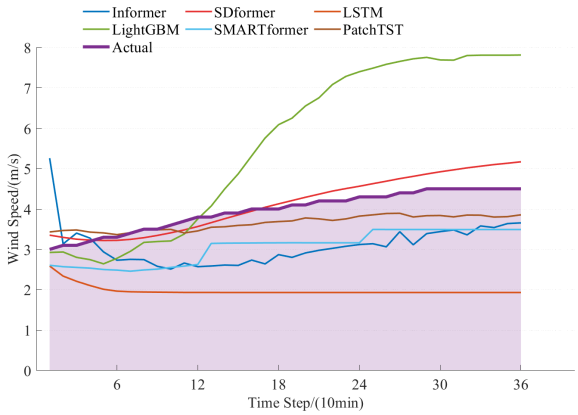

Fig. 3: Comparison of Prediction Results of Different Models for 6 - hour Forecasting

firm their individual contributions to performance. In 4-hour wind speed forecasting experiments (Table IV), both standalone modules showed performance drops: DeepResGCN's MSE/MAE increased by 17.6%/16.4% (0.173/0.318), while SMARTformer's MSE/MAE rose by 23.43%/17.53% (0.192/0.331) compared to the complete model. These results highlight the importance of combining both components to effectively capture wind speed spatiotemporal features.

E. Stability Analysis

To verify the stability of the proposed DSformer model, we conduct 100 independent training sessions. As shown in the figure, the distribution of MSE and MAE metrics on the test set demonstrates minimal fluctuations. This further confirms the high training stability of the DSformer model.

The MSE and MAE results of the DSformer model at different lead times (4 h, 6 h, and 8 h) indicate that the model's prediction error fluctuates little in the testing set, showing strong training stability and prediction accuracy. Particularly, the model performs stably and reliably in short - term predictions (4 h and 6 h).

IV. Conclusion

The experimental results demonstrate that our proposed DSformer model, based on DeepResGCN and SMARTformer, achieves high accuracy and stability in wind speed prediction. This is reflected in three aspects:

(1)The multi - graph feature fusion method we propose uses low - rank decomposition to reduce the

TABLE IV: Ablation Study Results of the Model's Main Components

Model Configuration	MSE	MAE
Complete Model	**0.147**	**0.273**
DeepResGCN	0.173	0.318
SMARTformer	0.192	0.331

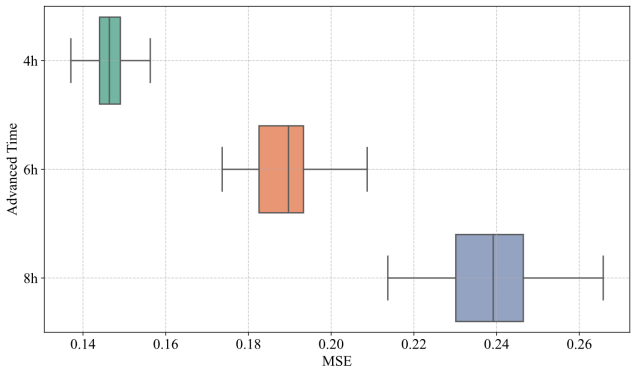

Fig. 4: MSE distribution in 100 independent experiments

dimensionality of multi - dimensional spatial - topological graph features, preserving key spatiotemporal associations. This enhances the representation ability of multi - graph structures in complex meteorological scenarios. Meanwhile, the dynamic feature fusion mechanism based on inverse reconstruction realizes the complementary integration of spatiotemporal features at different scales, improving the prediction stability of the model in typhoon - induced sudden wind speed changes.

(2)Our DSformer model shows superiority in spatiotemporal modeling and wind speed prediction. When predicting wind speeds at 4 - hour, 6 - hour, and 8 - hour intervals, it reduces MSE by 5.2%, 7.8%, and 11.3%, respectively.

(3)The model we designed shows high accuracy and stability in wind speed prediction, providing a valuable reference for further research in this field and significant support for enhancing wind power prediction performance.

In summary, our proposed wind speed prediction model and methodology are of great significance with broad application prospects in spatiotemporal wind speed prediction and positive impacts on wind power prediction. The combination of our model with wind turbine group models allows for direct application in wind power prediction, providing practical feasibility and valuable references for real - world applications.

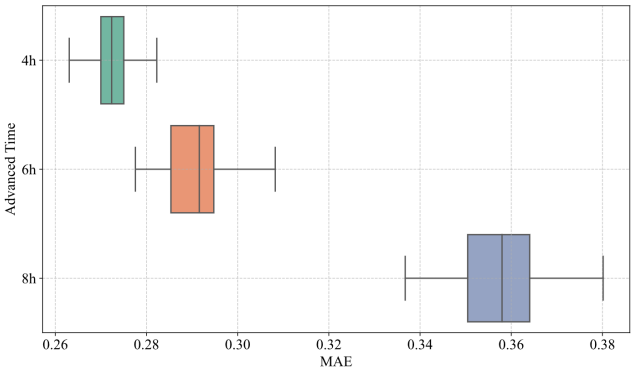

Fig. 5: MAE distribution in 100 independent experiments

References

[1] Flavio Corradini, Marco Gori, Carlo Lucheroni, Marco Piangerelli, and Martina Zannotti. A systematic literature review of spatio-temporal graph neural network models for time series forecasting and classification. *arXiv preprint arXiv:2410.22377*, 2024.

[2] Pei Du, Jianzhou Wang, Zhenhai Guo, and Wendong Yang. Research and application of a novel hybrid forecasting system based on multi-objective optimization for wind speed forecasting. *Energy Conversion and Management*, 150:90–107, 2017.

[3] Yiduo Li, Shiyi Qi, Zhe Li, Zhongwen Rao, Lujia Pan, and Zenglin Xu. Smartformer: Semi-autoregressive transformer with efficient integrated window attention for long time series forecasting. In *IJCAI*, pages 2169–2177, 2023.

[4] Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*, 2022.

[5] Alexander Sommers, Logan Cummins, Sudip Mittal, Shahram Rahimi, Maria Seale, Joseph Jaboure, and Thomas Arnold. A survey of transformer enabled time series synthesis. In *2024 IEEE 10th International Conference on Collaboration and Internet Computing (CIC)*, pages 60–69. IEEE, 2024.

[6] Wenxiao Wang, Wei Chen, Qibo Qiu, Long Chen, Boxi Wu, Binbin Lin, Xiaofei He, and Wei Liu. Crossformer++: A versatile vision transformer hinging on cross-scale attention. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):3123–3136, 2023.

[7] Yun Wang, Runmin Zou, Fang Liu, Lingjun Zhang, and Qianyi Liu. A review of wind speed and wind power forecasting with deep neural networks. *Applied energy*, 304:117766, 2021.

[8] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 11106–11115, 2021.

[9] 凡航, 张雪敏, 梅生伟, and 杨忠良. 基于时空神经网络的风电场超短期风速预测模型. *电力系统自动化*, 45(01):28–35, 2021.

[10] 向玲, 陈锦鹏, 付晓孟婷, and 姚青陶. 基于 vit 和 lstm 的风速多步预测. *太阳能学报*, 45(09):525–533, 2024.

[11] 姜兆宇, 贾庆山, and 管晓宏. 多时空尺度的风力发电预测方法综述. *自动化学报*, (01):51–71, 2019.

[12] 李敏. 基于聚类的风速混合预测模型. Master's thesis, 兰州大学, 2019.

[13] 杨茂, 陈新鑫, 张强, 李大勇, 孙涌, and 贾云彭. 基于支持向量机的短期风速预测研究综述. *东北电力大学学报*, (04):1–7, 2017.

[14] 王颖 and 魏云军. 风电场风速及风功率预测方法研究综述. *陕西电力*, 39(11):18–21+30, 2011.

[15] 袁咪咪, 宫法明, and 李昕. 基于 cnn-lstm 的多因素时空风速预测. *计算机系统应用*, 30(08):133–141, 2021.

[16] 闫帆 and 李傲燃. 基于集合经验模态分解和优化支持向量机的风速预测模型. *黑龙江电力*, 45(05):402–406, 2023.

[17] 黄宇, 张宗拾, 刘家兴, 李旭昕, and 张鹏. 基于改进时间卷积网络与藤 copula 的短期风速预测. *电力科学与工程*, 40(07):60–69, 2024.