ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 2 - 2026

Research on a wind power forecasting model for wind farm clusters based on the fusion of spatiotemporal graph convolutional networks and FedFormer

Xu Cao[1,*]

[1] Key Laboratory of Modern Power System Simulation and Control & Renewable Energy Technology, Ministry of Education (Northeast Electric Power University), Jilin 132012, China

SUMMARY: *Advances in artificial intelligence and intelligent algorithms have driven the evolution of wind power forecasting toward spatio-temporal collaborative modeling and distributed learning. This paper proposes a wind farm cluster power forecasting model that integrates ST-GCN and FedFormer. Based on wind farm cluster graph modeling, ST-GCN is used to extract spatially coupled and local temporal features, while FedFormer is employed to enhance the representation of long-term trends and frequency-domain information. Collaborative training is performed within a federated framework.Experimental results show that on the validation set, the model achieves MAE, RMSE, and MAPE of 16.87%, 23.14%, and 6.38%, respectively, outperforming FedFormer's 18.21%, 24.97%, and 6.95% and ST-GCN's 18.74%, 25.86%, and 7.21%, demonstrating higher accuracy and stability.*

KEYWORDS: *wind power forecasting; spatio-temporal convolutional network; FedFormer; federated learning; wind farm cluster*

1 Introduction

The progress of artificial intelligence and smart algorithms has fueled the development of wind power prediction beyond the classical statistical modelling to the deep spatiotemporal learning and the fusion of multi-source information. A single model can hardly solve all the problems of spatial coupling, long-term trend modeling, and engineering deployment needs due to the high non-stationarity, cross-site correlations, and long-term dependencies of power time series of wind farm clusters.

Bentsen et al. applied the concept of graph networks and new Transformer models to forecast spatiotemporal wind speeds, showing that a combination of graph-based modeling with attention mechanisms can be effective in improving spatiotemporal representation in complex situations [1]. Xu et al. presented the smart power forecasting of wind power based on smart city energy management, which highlights the significance of the multivariate perception and forecasting accuracy in sustainable scheduling [2]. A dual-branch spatiotemporal network that is suggested by Wei et al. and improves the performance of wind power forecasting by multimodal fusion has shown that the collaborative branch structure can be of great benefit in the extraction of heterogeneous features [3]. Xu et al. built the PatchTST-GRU heterogeneous sequence-to-sequence model and integrated it with numerical weather prediction to enhance results of multi-step forecasts, demonstrating the efficacy of combining long-sequence modeling with external prior information [4]. To further improve the capability of jointly

*2202400143@neepu.edu.cn
https://doi.org/10.65102/is2026816
Published: 30 April, 2026

describing local spatial correlations and dynamic dependencies, Wang et al. added spatio-temporal position attention to short-term power forecasting across multiple units [5]. Even though the current literature has achieved some advancements in graph learning, Transformer enhancements, dual-branch fusion, and multi-step forecasting, the vast majority of the studies are limited to centralized training models, and the issues of privacy limits, collaborative training, and cross-site generalization in the context of the distributed nature of the data of wind farm clusters have been underrepresented.

According to the foregoing discussion, the research motivation is to strengthen the modeling of space relationships, extracting long-sequence trends, and flexibility in distributed training in the forecasting of wind farms clusters. The areas of this work include the modeling of wind farm clusters graph, spatiotemporal feature extraction with ST-GCN, the long-sequence prediction branch of FedFormer, dual-branch fusion output, and the design of a federated training mechanism. Graph convolutional networks are used methodologically to describe spatial propagation relationships in wind farm clusters, FedFormer is used to improve frequency-domain trend representation, and multi-client collaborative optimization is applied in a federated structure. The anticipated result is a wind farm cluster power prediction scheme that balances accuracy of prediction, training effectiveness and privacy protection, offering a general guide to model layout, experimentation validation, and system analysis in the following parts.

2 Theoretical and Technical Foundations

2.1 Spatio-Temporal Graph Convolutional Network (ST-GCN)

Within the realm of predicting the power generation of clusters of wind farms, ST-GCN models the extraction of sequential fluctuations and the integration of multiple wind farms built through geographical proximity, power transmission pathways, and historical power correlations under a single construct. The spatial convolution component captures the changes in the strength of correlations between adjacent wind farms, enabling the effect of the wind farms in the upstream position disturbing the wind farm in the upstream position disturbing wind and being responded to by the downstream farms to be represented[1]. The temporal convolution component focuses on the continuous change of power and meteorological variables and aims to capture the evolving short-term, ramping, and local periodic variabilities. ST-GCN is used in the geospatial and temporal analysis for the front-end of the FedFormer set of branches extending the models. To demonstrate the mechanisms of spatial coupling and feature propagation wind farm clusters, ST-GCN is described in conjunction with a framework[2]. This framework integrates ST-GCN, the wind farms in the study area, and the various temporal and spatial connectivity coupling relationships in an upstream-downstream process, contributing towards the geospatial feature extraction and analysis. Figure 1 shows that case:

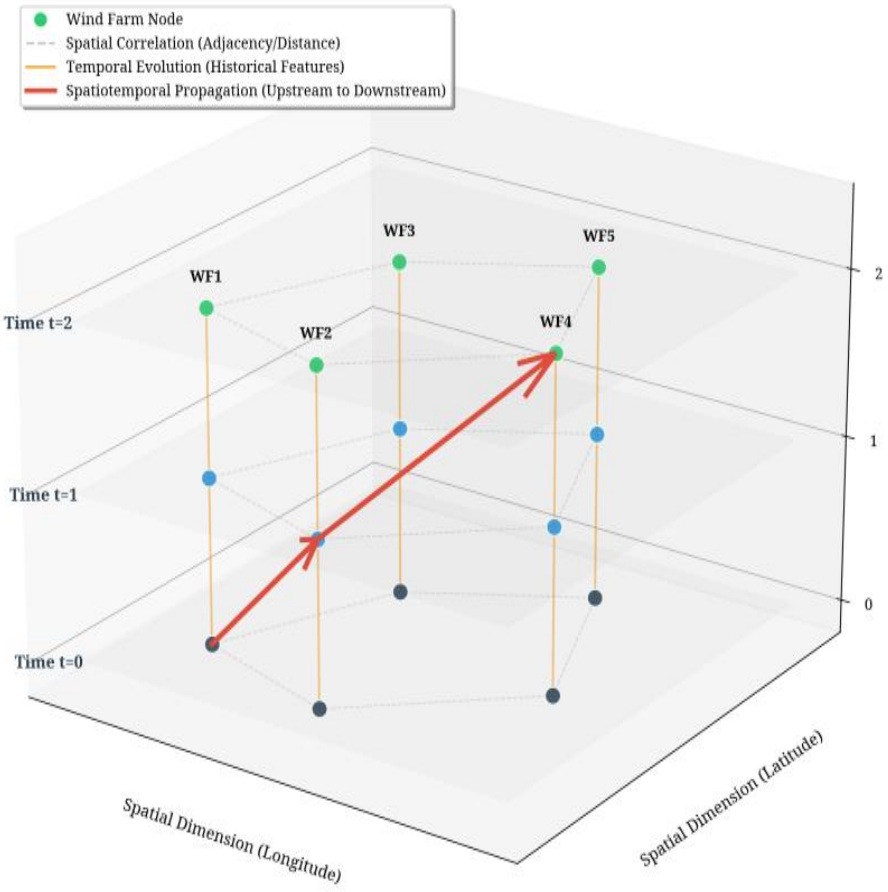

Figure 1: ST-GCN Spatio-Temporal Correlation Framework Diagram

2.2 Transformer Time Series Forecasting and FedFormer

The Transformer-based time series forecasting branch largely handles multivariate sequences concerning wind power and meteorological variables. It builds long-range gaps prime attention, and when used instead of recurrent models, is more flexible for multiple-step forecasting in a given time frame[3]. The project's core computations can be expressed as follows:

$$Att(Q,K,V) = soft\max\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{1}$$

where `$Q$` denotes the query matrix, `$K$` denotes the key matrix, `$V$` denotes the value matrix, `$T$` denotes the transpose, `$d$` denotes the feature dimension, `$soft\max$` denotes the normalization mapping, and `$Att$` denotes the attention output. This mechanism is used to measure the contribution of historical features at each time step to the prediction at the target time step[4].

Starting with that, FedFormer first sequences and, then long core models to align time series with the sliding-window input and multiple-step power output used in the following tests. The models' time series core decomposition can be expressed as follows:

$$x_t = \tau_t + s_t \tag{2}$$

where x_t denotes the raw input sequence at time t , t denotes the time index, τ_t denotes the trend component, and s_t denotes the seasonal component. In the experiments, τ_t corresponds to relatively stable long-term trends, while s_t corresponds to short-term fluctuations and random disturbances; together, they enhance the stability of long-sequence predictions[5].

2.3 Federated Learning Framework

The federated learning framework enables offline data sharing and supports collaborative training among independent wind farms. In the experiments, each client represents an independent wind farm or a regional node. After a certain number of local training steps, only the model weights will be uploaded to the server, and then a new round of global models will be distributed[6]. This is similar to the distributed prediction cases. The local update can be described as the following formula:

$$w_k^{r+1} = w^r - \eta \nabla F_k\left(w^r\right) \tag{3}$$

where w_k^{r+1} denotes the local model parameters obtained by the k th client in the r+1st round, w^r denotes the global model parameters in the r th round, η denotes the learning rate, ∇ denotes the gradient operator, F_k denotes the local loss function of the k th client, k denotes the client ID, and r denotes the communication round number. Server aggregation is performed using a weighted average:

$$w^{r+1} = \sum_{k=1}^{K} \frac{n_k}{n} w_k^{r+1} \tag{4}$$

where w^{r+1} denotes the global model parameters for the $r+1$ th round, $\sum$ denotes the sum over all clients, K denotes the total number of clients, n_k denotes the sample size of the k th client, n denotes the total sample size, and w_k^{r+1} denotes the local models uploaded by each client.

3 Design of the Integrated Prediction Model

3.1 Spatio-Temporal Graph Modeling of Wind Farm Clusters

The objective when applying spatio-temporal (ST) graph modeling techniques to wind farm clusters is to uniformly transform spatial relations and continuously recorded time-series data of wind farms into graph representations in order to establish a common data format for the initial phase of feature extraction in ST-GCN. Based on the experimental framework, each wind farm is represented as a node in the graph, with each node being characterized by historical power output and key variables of wind farm operational meteorology. Edge relations are arranged according the geographical proximity of the wind farms and the proximity of the wind farms in the clustering process is represented as:

$$G = (V, E, A) \tag{5}$$

where G is the spatio-temporal graph of the wind farm cluster, V is the set of nodes

representing all wind farm nodes; E is the set of edges representing the connections between wind farms; and A is the adjacency matrix used to describe the strength of the association between any two nodes[8].

At a given time t, the node characteristics of the i th wind farm can be expressed as:

$$x_{t,i} = \left[p_{t,i}, v_{t,i}, \theta_{t,i}, T_{t,i} \right] \tag{6}$$

Here, `$x_{t,i}$` represents the input feature vector of the `$i$` th node at time `$t$`, `$p_{t,i}$` represents wind power, `$v_{t,i}$` represents wind speed, `$\theta_{t,i}$` represents wind direction, `$T_{t,i}$` represents temperature, `$t$` represents the time index, and `$i$` represents the wind farm ID. This configuration aligns with the multivariate input window used in the experiments[9].

Spatial edge weights are constructed using a distance-decay method, specifically:

$$A_{ij} = \exp\left(-\frac{d_{ij}^2}{\sigma^2} \right) \tag{7}$$

where A_{ij} is the edge weight between nodes i and j, d_{ij} is the geographical distance between the two wind farms[10], σ is the distance decay coefficient, $\exp$ is the exponential function, and i and j represent the two different wind farm nodes, respectively.This methodology possesses the potential to capture the spatial dependencies of coupled wind farms in a cluster and aligns with the subsequent time-sliced predictions experiments.To further articulate the process of spatio-temporal graph modeling of wind farm clusters, examples of a modeling diagram and an adjacency matrix heatmap are included. The former shows the node and edge time sample construction, and the latter illustrates the gradation of wind farm proximity, thereby preparing for the initial phase of ST-GCN spatiotemporal feature extraction. As illustrated in Figure 2:

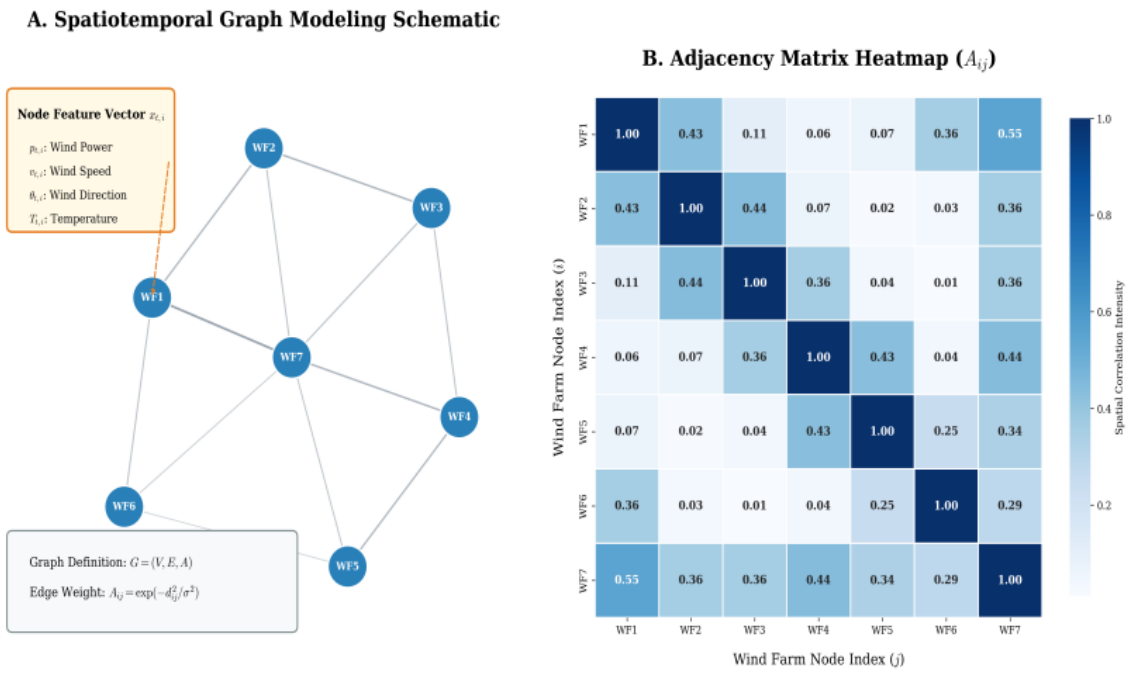

Figure 2: Modeling Diagram and Adjacency Matrix Heatmap

3.2 ST-GCN Spatio-Temporal Feature Extraction Module

The ST-GCN module's spatiotemporal features focus on standardizing the encoding process of the wind farm clusters' spatial correlation structure and their observations over time. This is modeled in a high-dimensional space and allows the extraction of spatial coupling characteristics and local temporal characteristics of power fluctuations[11]. The spatial dimension focuses on the correlation propagation, and the historical power correlation among the different wind farms. The temporal dimension corresponds to the continuous propelling of features over time in a sliding window. This can be expressed as:

$$H_t^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H_t^l W^{(l)} \right) \tag{8}$$

where $H_t^{(l)}$ denotes the node feature matrix at time t in layer l , $H_t^{(l+1)}$ denotes the output of the next layer, t denotes the time index, l denotes the network layer number, $\tilde{A}$ denotes the adjacency matrix after incorporating self-connections, $\tilde{D}$ denotes the degree matrix corresponding to $\tilde{A}$, $W^{(l)}$ denotes the trainable weight matrix for layer l , and σ denotes the nonlinear activation function. This process corresponds to the joint input of multiple wind farms in the experiment and enables the aggregation of neighborhood information[12].

The temporal convolution update can be expressed as:

$$Z_t = \sum_{k=0}^{K-1} \theta_k H_{t-k} \tag{9}$$

where Z_t denotes the temporal convolution output at time t , H_{t-k} denotes the input features from the k th time step prior to the current time, k denotes the temporal convolution kernel index, K denotes the temporal convolution kernel length, and θ denotes the convolution parameters corresponding to the k th time step. This calculation method aligns with the fixed-time-window input configuration used in the experiments, enabling the extraction of short-term fluctuations, ramp changes, and local periodic features, thereby providing a stable spatio-temporal representation for subsequent prediction branches[13].

3.3 FedFormer Time Series Forecasting Branch

The FedFormer time series prediction branch is designed to predict multiple wind forecast steps, given a fixed history. Its processing focus is on extracting stationary trends and compressing dominant frequency components of the cyclic fluctuation trends. This is applied to align the experiments settings of a sliding window input and multi-step output. Trend can be obtained through a process of local averaging:

$$\tau_t = \frac{1}{w} \sum_{i=0}^{w-1} x_{t-i} \tag{10}$$

where τ_t denotes the trend component at time t , x_{t-i} denotes the raw input value at time $t-i$,

w denotes the smoothing window length, i denotes the summation index within the window, and $\sum$ denotes the summation operation. Based on the trend term, the main frequency coefficients after trend removal can be further extracted:

$$c_f = \sum_{t=1}^{L} (x_t - \tau_t) \cos\left(\frac{2\pi f t}{L}\right) \tag{11}$$

where c_f denotes the coefficient of the th frequency component (f), L denotes the length of the input sequence, f denotes the frequency index, π denotes pi, cos denotes the cosine transform, and $x_t - \tau_t$ denotes the detrended fluctuation term. The forecast output is generated jointly from the trend information and the retained principal frequency coefficients:

$$\hat{y}_{t+h} = W_h[\tau_t, c_1, ..., c_M] + b_h \tag{12}$$

where $\hat{y}_{t+h}$ denotes the predicted power at time $t+h$, h denotes the prediction step size, W_h denotes the output layer weights at step h , $[\tau_t, c_1, ..., c_M]$ denotes the concatenated vector of the trend term and the first M frequency coefficients, M denotes the number of retained principal frequencies, and b_h denotes the bias term. The branch can preserve long-term dependencies and periodic information at a finite computational cost[14].

3.4 Dual-Branch Fusion and Output Mechanism

The dual-branch fusion and output mechanism maps spatiotemporal correlation features and long-term time-series frequency-domain features into prediction space. The ST-GCN branch emphasizes coupling mechanisms and local changes in the wind farm cluster, while the FedFormer branch emphasizes trends and periodic changes in historical data[15]. To enable all the features to participate in multi-step forecasting directly, the outputs of the dual branches are dimensionally aligned and weighted at the fusion stage. The form of the fusion process can be displayed as:

$$H_t^{fus} = \alpha H_t^{st} + (1-\alpha) H_t^{fd} \tag{13}$$

where H_t^{fus} represents the fused feature at time step t , H_t^{st} represents the spatio-temporal feature extracted by the ST-GCN branch at time step t , H_t^{fd} represents the time-series feature extracted by the FedFormer branch at time step t , α represents the fusion weight with a range from 0 to 1, and t represents the time index. This form facilitates the observation of the contributions of different branches to the prediction results in experiments.

After being mapped by the output layer, the fused features yield power prediction values for future time steps, expressed as:

$$\hat{Y} = H^{fus} W_o + b_o \tag{14}$$

where $\hat{Y}$ denotes the multi-step prediction results output by the model, H^{fus} denotes the fusion feature matrix, W_o denotes the output layer weight matrix, and b_o denotes the output layer bias

term. If the prediction step size is set to H , then $\hat{Y}$ corresponds to the power prediction values for the next H time steps. This output method is consistent with the sliding window input and multi-step power prediction evaluation settings used in the experiments, facilitating subsequent accuracy comparisons and ablation analysis[16].

4 Federated Distributed Training Architecture

4.1 Client-Server Communication Framework

The client-server communication framework structures the parameter exchange traffic among multiple wind farm nodes. It allows each client to access the global model, execute local training, and submit the results, along with other clients, within the same training iteration[17]. In this communication model, as soon as the clients start the process, the server sends each of them the same batch of starting values of the parameters for that specific training iteration. The relationship can be expressed as:

$$w_k^{(r,0)} = w^r, \quad k \in S^{(r)} \tag{15}$$

where $w_k^{(r,0)}$ denotes the initial model parameters received by client k at the start of round r , w^r denotes the global model parameters distributed by the server in round r , r denotes the communication round number, k denotes the client ID, $S^{(r)}$ denotes the set of clients selected for round r , and $\in$ denotes the set of clients belonging to the selected group. This configuration aligns with the multi-client parallel training process in the experiments, ensuring that all nodes perform local updates from the same starting point.

Communication overhead can be calculated based on the amount of data transmitted in both directions per round as follows:

$$C^{(r)} = \sum_{k \in S^{(r)}} \left(d_k^{(r)} + u_k^{(r)} \right) \tag{16}$$

where $C^{(r)}$ represents the total communication volume for round r , $\sum$ represents the sum of all clients participating in this round, $d_k^{(r)}$ represents the amount of model data sent by the server to client k , and $u_k^{(r)}$ represents the amount of parameter update data uploaded by client k . The experiments were conducted with control over the communication rounds, the client and model number, and the model size. In this context, this framework can realistically model the parameter exchange required by the distributed prediction model of a wind farm cluster[18].

Figure simulated a client-server framework for parameter interactions in federated training for wind farm clusters. The described framework displays how a server provides a global model to each client. Clients perform training using local data, and send server parameter updates. Training employs communication and collaboration, which characterizes federated learning. See Figure 3 for more details:

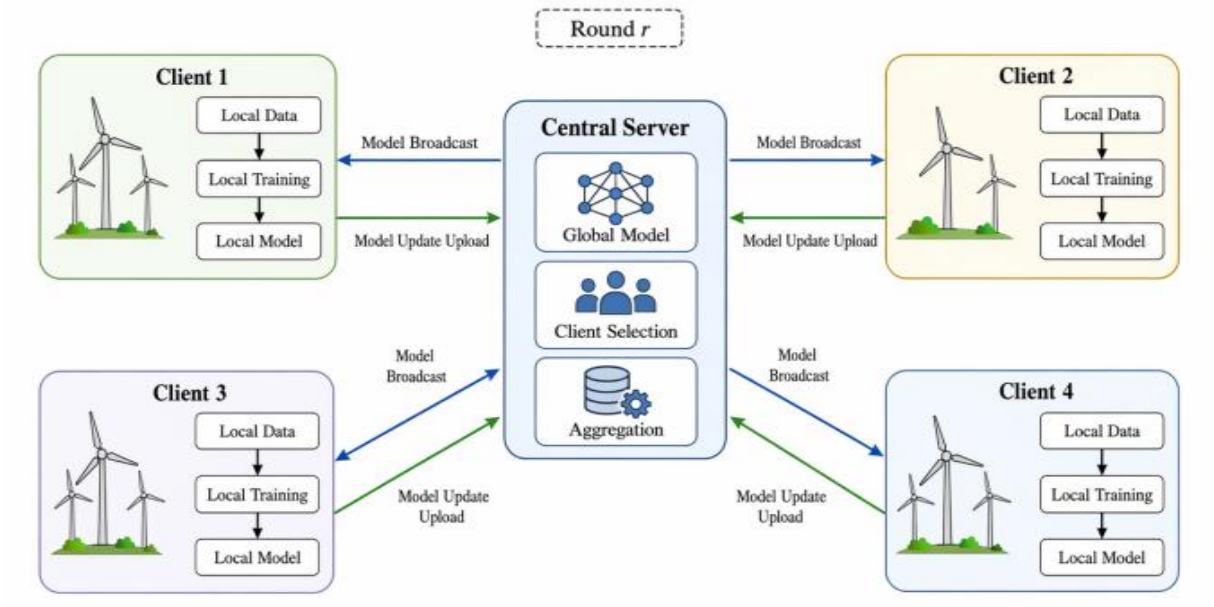

Figure 3: Client-Server Communication Framework Diagram

4.2 Local Update and Global Aggregation Strategies

The heterogeneity of the data among wind farm clients should be covered in the local update phase. Because various sites have very large differences in installed capacity, topography, meteorology distributions and power changes, customers are not supposed to do too many rounds of local iteration, because it is prone to overfitting the local data distributions;A more reasonable method is to update the spatially coupled features of ST-GCN and update the long-term sequence features of FedFormer equally in a fixed number of communication rounds[19]. In training, loss function and batch size must be normalized and validation set error must be kept in order to identify any drift in the local model. The global aggregation phase does not consist in the mere averaging but in the need to resolve the imbalance in parameters that occur when some clients have a greater sample size and/or a higher quality of data than others. Big stations can give more consistent statistical data, but when the pattern of fluctuation is monotonous, they can tend to suppress features elsewhere. Thus, when aggregation weights are to be considered, they must be modified according to local validation error or size of updates. This method maintains effective information of dominating wind fields and reduces the interference of abnormal updates with the global model and thus the aggregated model is more appropriate when joint forecasting is required among the wind fields.

4.3 Asynchronous Communication and Privacy Protection

Asynchronous communication in federated training of wind farm clusters: In federated training, prioritizing schedules that consider differences in client computing power, network latency, and data scale is important. When a rigid synchronization approach is still used, the edge node slow uploads will directly increase the global training cycle, making the server wait longer, and hence the efficiency of model iterations. Another more practical way is to enable clients to post parameters right after a local training, and the server does the aggregation in order of arrival or within a specified buffer time. Outdated model updates can be mitigated by timestamps, delay weights or aging decay. The method ensures that updates occur at high frequencies whilst ensuring that slow nodes do not consistently slow down the overall speed of training, and so is suitable in situations where there is a parallel integration of wind farms across geographical

locations. The idea of privacy protection should not be reduced to the principle of the data remaining local, because even the updates of the gradients and parameters can still indicate power distribution, the state of operation, the local meteorological features[20]. To overcome this risk, communication links must use both encrypted parameter send and secure aggregation, and the server only gets aggregated results but not plaintext updates sent by the individual clients. In the case of sensitive sites, noise can be pre-introduced before upload to cause the reverse engineering attacks to fail to reconstruct the power curves of individual sites. Meanwhile, the intensity of noise and compression ratio should not be too large, which would undermine the ability of the joint training of ST-GCN and FedFormer to differentiate parameters, resulting in a drop in prediction accuracy and convergence stability.

5 Experimental Validation and Results Analysis

5.1 Dataset and Experimental Environment

To simultaneously analyze the spatial correlation and temporal characteristics of the wind farm cluster. Available data must include power measurements and meteorological data, in a continuous format, for several wind farms. The data for the core attributes is to be organized based on a standard sampling time. Gaps will be filled, and data will be smoothed of outliers, and then the data will be normalized to set target points. Data will be divided based on time sequences to be tested, validated, and trained, and will not be shuffled. The system will be built to support distributed predictive and retroactive training. Federated Deep Learning (SDL) by means of PyTorch will be used for supported training. Graph-convolutional, long-sequence models, and parallel computing training phase iteration systems will be used to provide a basis for the accuracy and efficiency systems[21].

5.2 Comparison Methods and Evaluation Metrics

Before comparing the results later, it is first necessary to define the structure categories, training parameters and characteristics of the particular comparison model. The experimental conditions were uniformed with respect to model type, learning rate, batch size, parameter scale, and spatio-temporal modeling capacity, as indicated in Table 1;

Table 1: Comparison Methods and Experimental Settings

No.	Method Name	Core Architecture	Learning Rate	Batch Size	Number of Parameters (10,000)	Does it utilize spatial relationships?	Does it support long sequences?
1	HA	Historical average for the same period	—	—	0.00	No	No
2	LSTM	Two-layer LSTM	0.0010	64	18.7		General
3	GRU	Two-layer GRU	0.0010		14.2		
4	TCN	Residual Hollow Convolution	0.0008		11.6		Better
5	Transformer	Multi-head self-attention	0.0005	32	31.4		
6	ST-GCN	Spatial Convolution + Temporal Convolution	0.0010	32	26.9	Yes	General
7	FedFormer	Decomposition + Frequency Domain Modeling	0.0005	32	28.5	No	Yes
8	Fusion Model	ST-GCN+FedFormer	0.0005	32	39.8	Yes	

The methods of comparison, as illustrated in Table 1, include statistical models, recurrent networks, convolutional models, attention models, and graph-spatiotemporal models, and they illustrate a well-defined hierarchy. Of these, HA does not require training parameters and is appropriate as a baseline; LSTM and GRU have 187, 000 and 142, 000 parameters, respectively, which makes them less expensive to train; The Transformer model has 314, 000 parameters, and the hybrid model has 398, 000 parameters, which increase their ability to model long sequences. This is higher as compared to that of single models, but at the same time, it has the capability of modelling spatial relationships and process long sequences hence is better placed in the multivariate, multi-node forecasting of wind farm clusters[22].

When a single data partition is used and the same forecasting tasks are solved, the error and fitting of various methods on the validation set is a direct measure of the adaptability of the models. The discrepancies between the results of the compared methods in relation to mean absolute error, root mean square error, mean absolute percentage error and goodness of fit are summarized in Table 2:

Table 2: Statistical Results of Metrics for Each Comparative Method on the Validation Set

Method Name	Mean Absolute Error	Root Mean Square Error	Mean Absolute Percentage Error (%)	Goodness of Fit
HA	31.84	42.17	12.63	0.781
LSTM	22.46	30.91	8.97	0.892
GRU	21.88	29.74	8.56	0.901
TCN	20.95	28.67	8.14	0.910
Transformer	19.83	27.12	7.63	0.924
ST-GCN	18.74	25.86	7.21	0.936
FedFormer	18.21	24.97	6.95	0.943
Fusion Model	16.87	23.14	6.38	0.956

Table 2 shows the fusion model outperforming the rest in all four metrics. The MAE reduced considerably by 14.97 to 16.87 from HA's 31.84;the RMSE exhibited a reduction from 42.17 to 23.14, showing the steadiness in the representation of the peaks of the waves; the control of the relative deviation was improved, as the MAPE declined from 12.63% to 6.38%; and the fit improved from 0.936 and 0.943 in the case of ST-GCN and FedFormer, respectively, to 0.956, showing that the dual-branch structure was better suited to the combined modeling of spatiotemporal information.See Figure 4 for more details:

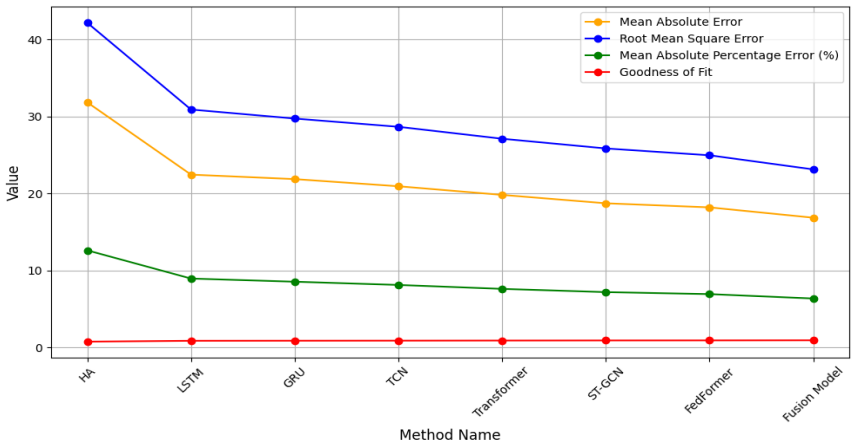

Figure 4: Performance Metrics of Different Methods

5.3 Analysis of Prediction Accuracy and Efficiency

In order to determine the stability of various models in multi-timescale forecasting tasks, one would need to compare further the error variations of various models to different forecast horizons. The test accuracy results of all the models in 15-minute, 30-minute, 60-minute and 120-minute forecasting tasks are summarized in table 3 and this could be used to analyze the performance degradation of the models with increasing forecast horizon. Table 3 demonstrates that:

Table 3: Comparison of Test Accuracy for Different Models at Various Forecast Intervals

Method Name	15 min MAE	15 min RMSE	30-Minute MAE	30-Minute RMSE	60-minute MAE	60-minute RMSE	120-minute MAE	120-minute RMSE
LSTM	15.92	21.84	18.37	25.66	22.94	31.48	28.76	39.22
TCN	15.34	20.97	17.85	24.58	21.86	29.94	27.45	37.10
Transformer	14.62	20.15	16.91	23.37	20.74	28.46	26.08	35.52
ST-GCN	13.88	19.41	16.03	22.56	19.67	27.38	24.95	34.11
FedFormer	13.54	18.96	15.71	22.03	19.18	26.74	24.36	33.28
Fusion model	12.76	17.98	14.83	20.92	18.05	25.43	22.97	31.86

As shown in Table 3, the errors of all models increase as the prediction horizon increases. However, the fusion model is the best at all levels of the horizon. The MAE and RMSE of the fusion model in the 15-minute time horizon are 12.76 and 17.98, as opposed to the losses of FedFormer of 13.54 and 18.96. The 120-minute time horizon saw the fusion model raise the the MAE and RMSE to 22.97, and the ST-GCN to RMSE of 34.11 and the Transformer to the RMSE of 31.86, a reduction of 5.79 and 3.11, respectively. This confirms that dual-branch models maintain strong trend modeling and error control for long-horizon tasks. More information is provided in Figure 5:

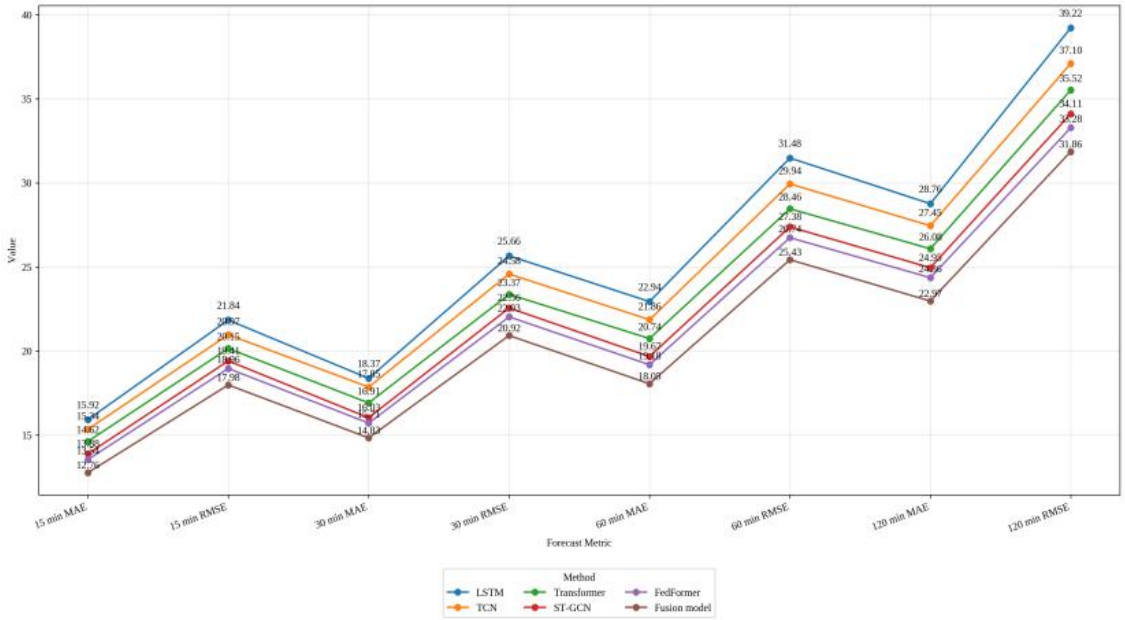

Figure 5: Comparison of Forecast Accuracy at Different Prediction Horizons

It's important to consider the real computational overhead of the models at the stages of training, communication, and inference when taking into account prediction accuracy. Table 4 provides a summary of the Most Efficient Model Performance in terms of single-episode training time, number of convergence episodes, total training time, single-sample inference time, single-episode communication volume, and GPU memory usage. As shown in Table 4:

Table 4: Comparison of Training and Inference Efficiency for Major Models

Method Name	Training Time per Iteration (s)	Number of Convergence Iterations	Total Training Time (min)	Inference Time per Sample (ms)	Data Transfer per Iteration (MB)	Peak VRAM usage (GB)
Transformer	22.8	46	17.5	3.1	11.8	3.6
ST-GCN	19.6	44	14.4	2.8	10.9	3.1
FedFormer	24.7	41	16.9	3.4	12.6	3.9
Fusion Model (Centralized)	29.8	39	19.4	4.2	—	4.5
Fusion Model (Federated Synchronization)	31.5	43	22.6	4.3	14.2	4.6
Federated Model (Asynchronous)	30.9	38	19.6	4.3	13.4	4.6

As shown in Table 4, compared to the single model method, the fusion model can be said to have a bigger model size, but the training time can be said to be reasonably fair. For the fully centralized fusion model, training took 19.4 minutes, which is only 5.0 minutes longer than training the ST-GCN in the federated model, which took 14.4 minutes. The federated model, with a synchronous approach, training took 22.6 minutes, while in the federated model, with the asynchronous approach, training took 19.6 minutes, which is 3.0 minutes shorter than the federated synchronous training. The single sample inference time is constant at 4.3 ms, which is less than the single sample inference time of the Transformer, which is 3.1 ms. The Transformer architecture is less than the 3.1 ms, but the performance metrics show better accuracy and performance, which is a good fair trade.See Figure 6 for more details:

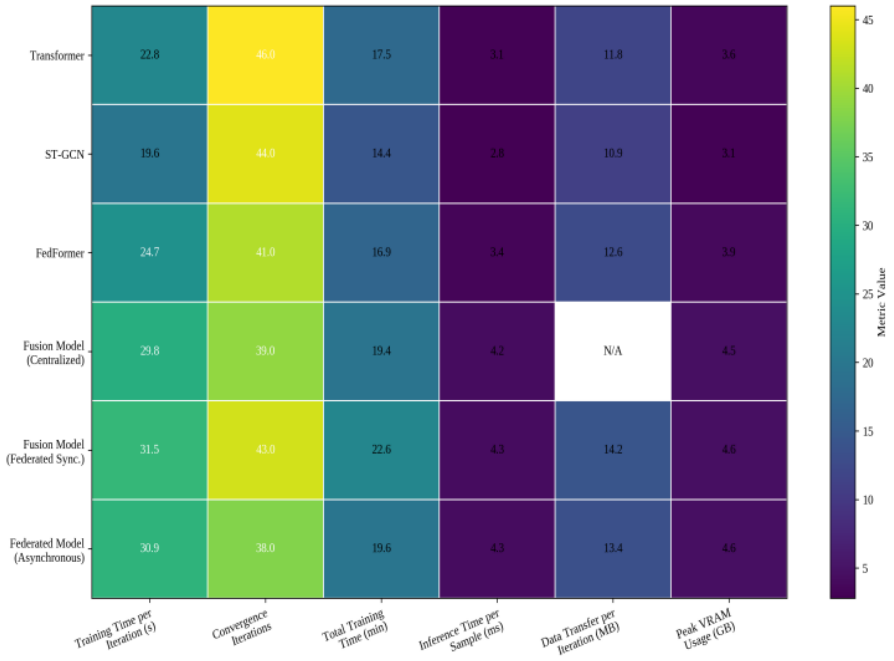

Figure 6: Heatmap of Training and Inference Efficiency Metrics

5.4 Ablation Studies and Visualization

Experiments with ablation ought to be performed systematically around the dual-branch architecture, federated training mechanism, and the important modeling components. To begin with, the control scenarios that we established include: eliminating the ST-GCN branch, eliminating the FedFormer branch, using feature fusion with a single-layer linear mapping, and federated training with independent training on a single client. We then compared error variations at the same data partition, prediction stride and training iterations to determine the performance improvements by spatial coupling modeling, long-sequence modeling, and distributed collaboration respectively. More sensitivity analyses may be performed on construction of adjacency matrices, fusion weight combinations, and local iteration numbers to see how the model is stable to different spatial connection weights and frequency of communication and thus not use one optimal solution to determine the validity of the model. The visualization part is not to present curves but interpret the results. To begin with, provide a comparison between actual and predicted values, paying attention to the necessity to adjust discrepancies in ramp-up parts, ramification peaks, and low-power periods to prove the responsiveness of the fusion model in complicated zones of variations. Second, the distribution plots of plot error or box plot to compare the dispersion of each model over time, depicting the difference in stability. The effectiveness of spatio-temporal modeling can be evaluated by displaying an adjacency matrix heatmap or node association weight map to determine information propagation routes between wind farms. In the case of the federated training process, convergence curves are provided in order to compare the number of iterations and the rate of error reduction using both the synchronous and asynchronous mechanisms, thus developing a relationship between the benefits of the models and the training properties.

6 Conclusions

We built a dual-branch forecasting framework that integrates ST-GCN and FedFormer to resolve issues with the spatial coupling relationships, long-term sequence dependencies, and cross-site data sharing challenges in wind farm power forecasting. Using a novel federated distributed training mechanism, we accomplished a unified ST-GCN modeling of spatio-temporal correlation features and FedFormer modeling of frequency-domain trend features, while maintaining privacy constraints. The results show that this novel approach predicts multiple time steps beyond the current horizon while maintaining privacy. The model Also, the asynchronous communication strategy improved training efficiency to some extent. The main contributions of this research show that the proposed method beyond the current horizon is threefold: i) graph-structured representations of wind farm clusters, ii) a novel dual-branch feature integration method, and iii) a novel method of federated collaborative optimization. The construction of adjacency relationships remains partially static, the impact of client data heterogeneity on global aggregation has not been addressed, and there is a trade-off between privacy and prediction performance. In that sense, we suggest in the future to work on dynamic graphs, more advanced adaptive aggregation, and lightweight joint multi-energy forecasting.

References

[1] Bentsen L Ø, Warakagoda N D, Stenbro R, et al. Spatio-temporal wind speed forecasting using graph networks and novel Transformer architectures[J]. Applied Energy, 2023, 333: 120565.

[2] Xu Z, Kong Y, Shen A. Intelligent Wind Power Forecasting for Sustainable Smart Cities[J]. Applied Sciences, 2025, 16(1): 305.

[3] Wei J, Zhang W, Zhang W, et al. DBSTN: A dual-branch spatio-temporal network for wind power prediction using multi-modal fusion[J]. Energy, 2025: 139471.

[4] Xu S, Wang Y, Xu X, et al. A PatchTST-GRU-based heterogeneous seq2seq model with numerical weather prediction refinement for multi-step wind power forecasting[J]. Scientific Reports, 2025, 15(1): 16547.

[5] Wang Q, Si G, Qu K, et al. Integrating spatio-positional series attention into deep networks for multi-turbine short-term wind power prediction[J]. Journal of Renewable and Sustainable Energy, 2024, 16(1).

[6] Meng W, Sun P, Yan Y, et al. Enhanced offshore wind power forecasting through multiscale time series decomposition and temporal pattern attention mechanisms[J]. International Journal of Green Energy, 2026, 23(1): 110-128.

[7] Wu J, Chang Z, Zhang L, et al. A dual-branch transformer network with multi-scale attention mechanism for microgrid wind turbine power forecasting[J]. Electronics, 2025, 14(13): 2566.

[8] Ke Y, Liu X, Ge-Zhang S, et al. Efficient wind power prediction via hybrid preprocessing and enhanced Reformer[J]. International Journal of Electrical Power & Energy Systems, 2026, 176: 111694.

[9] Yang Y, Lou H, Wu J, et al. A survey on wind power forecasting with machine learning approaches[J]. Neural Computing and Applications, 2024, 36(21): 12753-12773.

[10] Qin Y, Hu J, Qi H, et al. A Short-term Wind Power Forecasting Model Based on Trend-Seasonal Decoupling and Dual-branch Heterogeneous Modeling[C]//2025 6th New Power System International Forum-Power System and New Energy Technology Innovation Forum (NPSIF). IEEE, 2025: 368-377.

[11] Lv Q, Zhang Y, Wu G, et al. CSAT-Former: A Cross-Scale Aligned Transformer for Hierarchical Wind Power Forecasting with Temporal Consistency[C]//2025 International Conference on Power Systems, Smart Grid, and Artificial Intelligence (PSGAI). IEEE, 2025: 53-60.

[12] Zhuang W, Li Z, Wang Y, et al. GCN–Informer: A novel framework for mid-term photovoltaic power forecasting[J]. Applied Sciences, 2024, 14(5): 2181.

[13] Ay A, Önal K, Top A, et al. Comparative Deep Learning Models for Short-Term Wind Power Forecasting: A Real-World Case Study from Tokat Wind Farm, Türkiye[J]. Symmetry, 2025, 18(1): 11.

[14] Huang Q, Wang Y, Yang X, et al. Research on wind power prediction based on a gated transformer[J]. Applied Sciences, 2023, 13(14): 8350.

[15] Li X, Tang G. Multivariate sequence prediction for graph convolutional networks based on ESMD and transfer entropy[J]. Multimedia Tools and Applications, 2024, 83(35):

83493-83511.

[16] Xing F, Gao Y, Kang L, et al. KAN-Transformer model for ultra-short-term wind power prediction based on EWMA data processing[J]. Applied Sciences, 2024, 14(21): 9630.

[17] Tie R, Li M, Zhou C, et al. Research on the application of an improved Autoformer model integrating CNN-attention-BiGRU in short-term power load forecasting[J]. Evolving Systems, 2025, 16(3): 98.

[18] Suresh V. Benchmarking Transformer variants for hour-ahead PV forecasting: PatchTST with adaptive conformal inference[J]. Energies, 2025, 18(18): 5000.

[19] Lv Y, Hu Q, Xu H, et al. An ultra-short-term wind power prediction method based on a spatio-temporal attention graph convolutional model[J]. Energy, 2024, 293: 130751.

[20] Tang J, Liu Z, Hu J. Spatio-temporal wind power probabilistic forecasting based on time-aware graph convolutional network[J]. IEEE Transactions on Sustainable Energy, 2024, 15(3): 1946–1956.

[21] Dong X, Sun Y, Li Y, et al. Spatio-temporal convolutional network-based power forecasting of multiple wind farms[J]. Journal of Modern Power Systems and Clean Energy, 2021, 10(2): 388-398.

[22] Li Z, Ye L, Zhao Y, et al. A spatiotemporal directed graph convolution network for ultra-short-term wind power prediction[J]. IEEE Transactions on Sustainable Energy, 2022, 14(1): 39-54.