International Journal of Scientific Engineering and Research (IJSER)
ISSN (Online): 2347-3878
SJIF (2025): 8.036

Short-Term Wind Power Forecasting Based on a BiLSTM-Transformer Hybrid Architecture

Mingjia LV

North China Electric Power University, School of Control and Computer Engineering,
No. 2 Beinong Road, Changping District, Beijing, China
Email: *2500628722[at]qq.com*

Abstract: ***High-accuracy short-term wind power forecasting is essential for stable power grid operation. This paper proposes a BiLSTM-Transformer hybrid framework for time series prediction to address wind power volatility. The method leverages deep learning by integrating refined feature engineering (anomaly detection and PCC) with a dual-channel architecture. It captures local temporal dynamics via BiLSTM and long-range dependencies via Transformer, followed by an adaptive feature fusion mechanism. Experiments on a real-world SCADA dataset yield an MAE of 181.24 kW, RMSE of 202.58 kW, MAPE of 6.16%, and a coefficient of determination of of 0.916. Compared to standalone LSTM and Transformer models, the framework reduces RMSE by 24.5% and 11.4%, respectively, demonstrating superior performance in short-term forecasting.***

Keywords: Wind Power Forecasting; BiLSTM–Transformer; Deep learning; Short-term forecasting; Time Series Prediction; Feature Fusion

1. Introduction

Against the backdrop of the intensifying global energy crisis and the continued advancement of the “dual carbon” strategy, the energy structure is undergoing a transition toward clean and low-carbon sources. Wind power, with its renewability, technological maturity, and environmental advantages, has gradually become a key component of modern energy systems, playing an important role in enhancing energy security and mitigating climate change.

However, wind power output is highly dependent on meteorological and geographical conditions, such as wind speed, wind direction, temperature, and terrain, resulting in strong randomness, volatility, and intermittency. This uncertainty not only complicates accurate power prediction but also increases the challenges of grid operation, including peak regulation and power balancing. Therefore, improving the accuracy of wind power forecasting is essential for reducing grid integration risks, enhancing renewable energy utilization, and ensuring the safe and reliable operation of power systems.

2. Related Work

Accurate wind power prediction is fundamental for supporting economic dispatch in modern power systems. In recent years, data-driven deep learning methods have shown strong capability in handling the nonlinearity and volatility of wind power, gradually replacing traditional statistical approaches.

Sun Rongfu et al. (2021) highlighted that wind power fluctuations increase the pressure on reserve capacity allocation, making accurate forecasting essential for grid security [1]. Han Zifen et al. (2019) reviewed forecasting methods and pointed out the limitations of traditional physical and statistical models in capturing nonlinear dynamics [2]. To improve data quality, Li Junqing et al. (2019) proposed a feature selection method based on data mining, effectively reducing redundant meteorological information [3]. Qi Fangzhong et al. (2025) further enhanced prediction performance for non-stationary data through multi-feature fusion and variational mode decomposition [4].

Deep learning models, especially Long Short-Term Memory (LSTM) networks and their variants, have demonstrated strong performance in time series forecasting. Fu Yang et al. (2022) [5] proposed an LSTM-TCN model to capture local fluctuation features, while Dhaka et al. (2025) [6] combined feature engineering with attention mechanisms to improve prediction accuracy. However, RNN-based models still suffer from long-term information degradation.

To address long-sequence modeling limitations, Transformer-based methods have been widely introduced. Luo Zhao et al. (2023) [7] proposed a multi-scale Transformer to capture long-range dependencies, and Zhang Yali et al. (2024) [8] developed a non-stationary Transformer model to reduce prediction lag under severe fluctuations. Pan et al. (2022) [9] demonstrated the effectiveness of a spatiotemporal graph Transformer in complex feature fusion. In addition, Li et al. (2025) [10] proposed a GCN-BiLSTM hybrid model, emphasizing the advantages of multi-model collaboration in improving prediction robustness.

In summary, although significant progress has been achieved in feature engineering, local feature extraction using BiLSTM, and global modeling with Transformer, a single model still struggles to simultaneously capture sharp local meteorological fluctuations and stable long-term global trends. To address this issue, this paper proposes a short-term wind power forecasting method for a single site based on refined feature

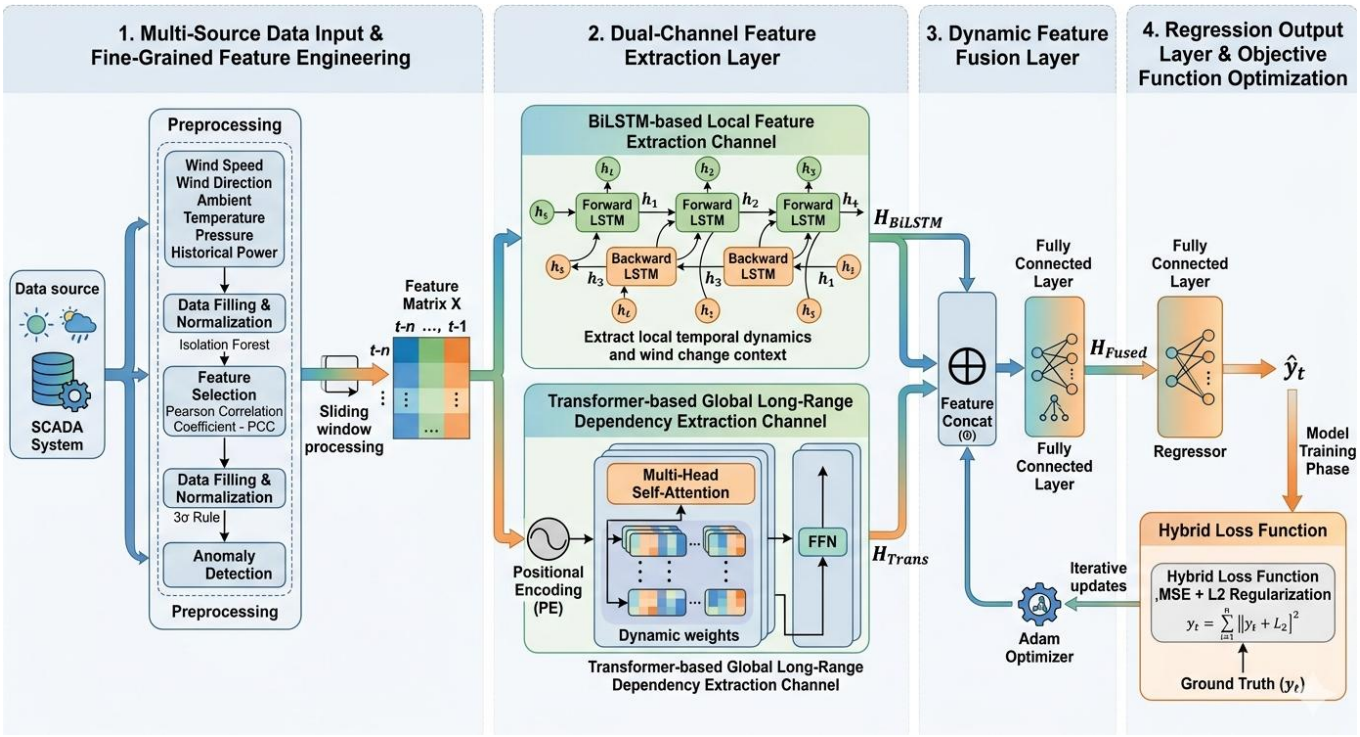

Figure 1: BiLSTM–Transformer Hybrid Framework engineering and a BiLSTM–Transformer hybrid architecture. The main contributions of this study are as follows:

1) A refined multidimensional feature engineering scheme is proposed, where noisy meteorological and power data are standardized and key features are selected to reduce redundancy and improve input quality.
2) A hybrid feature extraction architecture combining BiLSTM and Transformer is developed. BiLSTM captures short-term local fluctuations, while the Transformer models long-range dependencies, enabling effective fusion of local and global features.
3) Extensive experiments on real wind farm data show that, for 1–6 hour ahead forecasting, the proposed model achieves MAPE within 7% and outperforms conventional deep learning models in both accuracy and robustness.

3. Prediction Model and Methodology

Wind power output is highly influenced by meteorological conditions, and its time series not only exhibits intense short-term local fluctuations but also contains long-term global evolution trends. To comprehensively capture these complex characteristics, a short-term wind power forecasting method based on a BiLSTM-Transformer hybrid architecture is developed. This method mainly consists of four core modules. These modules include refined feature engineering, local temporal feature extraction based on BiLSTM, global long-range dependency modeling based on Transformer, and dynamic feature fusion for the final output. The overall architecture of the proposed BiLSTM-Transformer-based wind power forecasting model is illustrated in Figure. 1.

3.1 Refined Feature Engineering and Preprocessing of Meteorological Data

In practical wind farm SCADA (Supervisory Control and Data Acquisition) systems, the collected raw meteorological data (such as wind speed, wind direction, temperature, and air pressure) and power data are often accompanied by missing values and abnormal noise. Directly feeding such data into the model may lead to error amplification. Therefore, it is essential to perform refined feature engineering on multi-source heterogeneous data.

3.1.1 Outlier Detection and Missing Value Imputation

To address abnormal zero values and outliers caused by sensor failures or communication interruptions, a combined approach integrating the 3σ rule (Pauta criterion) and the Isolation Forest algorithm is adopted for anomaly detection and removal. For the resulting missing segments, linear interpolation is applied when the missing duration is short (e.g., less than three time steps). For longer missing intervals, values are replaced with historical averages under similar meteorological conditions to ensure the continuity of the time series.

3.1.2 Feature Normalization

Due to significant differences in scale and value distribution among wind speed, air pressure, temperature, and wind power, Min–Max normalization is applied to map all input features into the range of [0, 1], thereby accelerating neural network convergence and eliminating the influence of differing units. The calculation formula is given as follows:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Where, x represents the original data, x_{max} and x_{min} are the maximum and minimum.

3.1.3 Core Feature Selection

To reduce the dimensionality burden caused by redundant meteorological features, the Pearson Correlation Coefficient (PCC) is employed to analyze the correlation between meteorological variables and wind power. The most highly

correlated core features- such as hub-height wind speed, sine/cosine components of wind direction, and ambient temperature- are selected to construct a high-quality multidimensional input feature matrix $X \in R^{T\times D}$, where T denotes the number of time steps and D represents the feature dimension.

3.2 BiLSTM-Based Local Temporal Feature Extraction Module

Long Short-Term Memory (LSTM) networks, by incorporating forget, input, and output gates, effectively mitigate the gradient vanishing problem of traditional RNNs. However, standard LSTM relies solely on forward sequential information and cannot leverage future contextual states. To more finely capture the short-term local fluctuation patterns of wind power, the model further employs a BiLSTM network.

A BiLSTM consists of two independent LSTM layers: a forward layer and a backward layer. At a given time step t, the forward LSTM computes the hidden state $\overrightarrow{h_t}$ from past to future, while the backward LSTM computes the hidden state $\overleftarrow{h_t}$ from future to past. This can be mathematically expressed as:

$$\overrightarrow{h_t} = \mathrm{LSTM}_{forward}(x_t, \overrightarrow{h_{t-1}}) \quad (2)$$

$$\overleftarrow{h_t} = \mathrm{LSTM}_{backward}(x_t, \overleftarrow{h_{t+1}}) \quad (3)$$

The hidden states from the forward and backward directions are then concatenated to obtain the final local temporal feature representation H_t^{BiLSTM}:

$$H_t^{BiLSTM} = [\overrightarrow{h_t} \oplus \overleftarrow{h_t}] \quad (4)$$

Where, $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ represent the hidden features extracted by the forward and backward LSTM at time step t, respectively, and $\oplus$ denotes concatenation along the feature dimension.

This bidirectional mechanism fully captures contextual dynamics surrounding abrupt wind speed changes, enabling effective extraction of high-frequency local fluctuations.

3.3 Transformer-Based Global Long-Range Dependency Extraction Module

To overcome the computational bottlenecks and long-range information decay inherent in recurrent networks when processing very long sequences, a Transformer encoder module is introduced, leveraging the self-attention mechanism to compute global dependencies across the sequence in parallel.

3.3.1 Positional Encoding

Since the Transformer removes recurrent structures, absolute temporal information must be injected into the sequence using sinusoidal functions as positional encodings:

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{2i/d_{model}}}\right) \quad (5)$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{2i/d_{model}}}\right) \quad (6)$$

Where, pos denotes the time step position, i is the dimension index, and d_{model} represents the feature dimension. The input matrix is then updated as $X_{PE} = X + PE$.

3.3.2 Multi-Head Self-Attention Mechanism and Residual Network

The self-attention mechanism linearly projects the input into the query matrix Q, key matrix K, and value matrix V:

$$\mathrm{Attention}(Q,K,V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (7)$$

Where, Q, K K, and V are obtained by projecting the input sequence through separate linear transformations, and are used to compute the attention weights between different time steps. The division by $\sqrt{d_k}$, where d_k is the dimension of the key matrix K, prevents excessively large dot-product values that could destabilize gradients.

By employing Multi-Head Attention, the outputs of several independent attention heads are concatenated and then linearly projected, enabling the model to capture long-range dependencies across different subspaces:

$$MultiHead = Concat(head_1, head_2, ..., head_h)W^O \quad (8)$$

Where, $head_h$ denotes the output of the h-th independent attention head, and W^O is the weight matrix for the final linear projection.

Subsequently, residual connections and layer normalization are applied, effectively preventing the degradation problem in deep networks:

$$H_{sub} = LayerNorm(X_P E + MultiHead(X_P E, X_P E, X_P E)) \quad (9)$$

Where, H_{sub} denotes the intermediate sublayer feature matrix, and $LayerNorm$ is applied to stabilize the data distribution in deep networks and accelerate model convergence.

3.3.3 Feedforward Neural Network (FFN)

To enhance the model's nonlinear representation capability, a feedforward neural network consisting of two linear layers and a ReLU activation function is added, followed by another residual connection:

$$\mathrm{FFN}(H^{sub}) = \max(0, H^{sub}W_1 + b_1)W_2 + b_2 \quad (10)$$

$$H^{Trans} = \mathrm{LayerNorm}(H^{sub} + \mathrm{FFN}(H^{sub})) \quad (11)$$

Where, W_1, b_1 and W_2, b_2 denote the weight matrices and bias vectors of the two linear layers within the FFN, respectively; H^{Trans} represents the final output of the Transformer module, containing deeply encoded global long-range dependencies.

3.4 BiLSTM–Transformer Hybrid Architecture and Objective Function Optimization

To capture both local fluctuation details and global temporal trends, a dual-channel parallel hybrid architecture is designed. The preprocessed high-dimensional feature matrix x is simultaneously fed into the BiLSTM and Transformer channels. After extracting local features H^{BiLSTM} and global features H^{Trans}, the features are concatenated and dynamically fused through a fully connected layer:

$$F_{fusion} = \mathrm{ReLU}\left(W_f \cdot [H^{BiLSTM} \oplus H^{Trans}] + b_f\right) \quad (12)$$

Finally, the fused features are fed into a regression output layer to obtain the predicted wind power values $\hat{y}_t$ for future time steps:

$$\hat{y}_t = W_{out} \cdot F_{fusion} + b_{out} \quad (13)$$

Where, F_{fusion} represents the high-dimensional fused feature representation; W_f, b_f and W_{out}, b_{out} denote the weight matrices and bias vectors of the feature fusion layer and the final regression output layer, respectively; $\hat{y}_t$ is the predicted wind power value at time step t.

During the model training phase, to improve prediction stability and prevent overfitting, the Mean Squared Error (MSE) combined with an L2 regularization term (weight decay) is adopted as the overall objective loss function $L(\theta)$:

$$L(\theta) = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 + \lambda \| \theta \|_2^2 \quad (14)$$

Where, N denotes the batch size used for a single model parameter update; y_i and $\hat{y}_i$ represent the observed and predicted wind power for the i -th sample, respectively; θ is the set of all learnable parameters in the deep network (including all weights W and biases b); λ is the L2 regularization coefficient, a hyperparameter controlling the penalty on complex model weights; and $\| \theta \|_2^2$ is the squared L2 norm of the parameters. The L2 term constrains the scale of network weights, effectively preventing overfitting and enhancing the model's generalization ability to unseen meteorological data. The network parameters are optimized iteratively via gradient computation and backpropagation using the Adam optimization algorithm.

4. Experiments and Results Analysis

To validate the effectiveness and superiority of the proposed BiLSTM–Transformer hybrid architecture for short-term wind power forecasting, comprehensive comparative experiments were conducted using real-world wind farm SCADA datasets. The analysis examined multiple aspects, including prediction accuracy, model fitting performance, and error distribution.

4.1 Dataset Description and Experimental Environment

4.1.1 Dataset Description

The experiments utilized the Wind Turbine SCADA Dataset, which records the 2018 operational status of a single wind turbine in Turkey at 10-minute intervals (50,530 samples). To ensure clear methodology and prevent data leakage, the experimental setup is structured as follows:

- **Input Features:** Actual wind speed (m/s), wind direction (°), and historical active power (kW). The theoretical power curve variable was excluded.
- **Data Partitioning:** Chronologically split into training (70%, ~35,000 samples), validation (10%), and test sets (20%, ~10,000 samples).
- **Sequence Settings:** The input sequence spans 24 times steps (past 4 hours) to predict the subsequent 6 steps (upcoming 1 hour).

4.1.2 Experimental Environment and Parameter Settings

The model was implemented using Python 3.8 and the PyTorch deep learning framework. The hardware environment consisted of an Intel Core i7 processor and an NVIDIA RTX 4060 GPU with 8 GB of memory. In terms of hyperparameter settings, the BiLSTM module was configured with a hidden dimension of 64 and 2 layers, while the Transformer encoder employed 4 attention heads and a feedforward network dimension of 128. The overall model was optimized using the Adam optimizer with an initial learning rate of 0.001 and a batch size of 64. An early stopping mechanism was also introduced, whereby training was terminated if the validation loss did not decrease for 15 consecutive epochs, in order to retain the optimal model weights.

4.2 Evaluation Metrics

To objectively and quantitatively evaluate the performance of the forecasting model, three core metrics are adopted: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE). The calculation formulas for these metrics are given as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \quad (15)$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \quad (16)$$

$$MAPE = \frac{100\%}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \quad (17)$$

Where, N denotes the total number of samples in the test set; y_i represents the true observed wind power of the i -th sample; and $\hat{y}_i$ is the corresponding predicted value generated by the model. MAE and RMSE directly reflect the absolute magnitude of prediction errors, where smaller values indicate lower deviation. MAPE, on the other hand, measures the relative error, providing important insight into prediction stability, particularly under conditions of extreme fluctuations.

4.3 Comparative Experiments and Results Analysis

To highlight the advantages of the proposed BiLSTM–Transformer hybrid architecture, three classical baseline models were selected for comparative experiments, including

Long Short-Term Memory (LSTM), a hybrid Convolutional Neural Network–LSTM model (CNN–LSTM), and a standalone self-attention model (Transformer). The average prediction error results of each model for forecasting the next 1 hour (6 time steps) on the test set are summarized in Table 1.

Table 1: Error Comparison of Different Forecasting Models on the Test Set

Forecasting Models	*MAE (kW)*	*RMSE (kW)*	*MAPE (%)*
Single LSTM	235.60	268.42	10.84%
CNN-LSTM	212.35	241.15	9.15%
Single Transformer	203.45	228.62	8.42%
BiLSTM-Transformer	181.24	202.58	6.16%

The quantitative results in Table 1 reveal the limitations of baseline models and highlight the superiority of the proposed architecture:

- **Baseline Limitations:** Standalone LSTM struggles with high-frequency fluctuations, yielding the highest errors (MAE: 235.60 kW, MAPE: 10.84%). While CNN-LSTM slightly improves local feature extraction (MAPE: 9.15%), and the Transformer better captures global dependencies (MAPE: 8.42%), neither effectively handles both short-term volatility and long-term trends simultaneously.
- **Proposed Model Superiority:** The BiLSTM-Transformer hybrid achieves the lowest errors across all metrics (MAPE: 6.16%). Compared to the standalone LSTM and Transformer, it reduces RMSE by 24.5% and 11.4%, respectively. This confirms that fusing BiLSTM's local temporal extraction with the Transformer's global dependency modeling successfully decodes complex, non-stationary meteorological patterns.

4.4 Prediction Curves and Fitting Analysis

To more intuitively demonstrate the prediction and tracking capabilities of each model under complex working conditions, a portion of continuous wind power time series slices were extracted from the test set for visualization analysis, as shown in Figure 2.

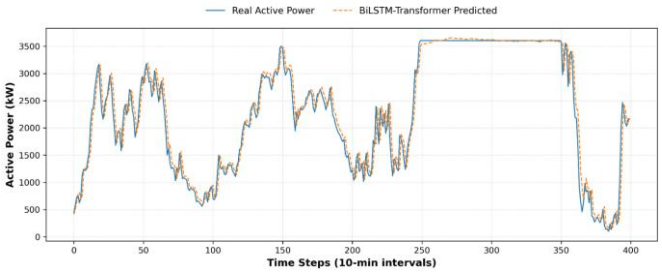

Figure 2: Comparison of Predicted and Actual Wind Power Time Series

From the prediction curve, it can be observed that during stable output periods, all deep learning models achieve reasonably good fits. However, in regions of strong fluctuations caused by abrupt wind speed changes- resulting in sudden "spikes" or "cliff-like drops" in power—the baseline models, such as standalone LSTM and CNN–LSTM, tend to exhibit noticeable peak lag and amplitude underestimation. In contrast, the proposed BiLSTM–Transformer hybrid model, leveraging the Transformer's multi-head self-attention mechanism for precise localization of critical meteorological change points and BiLSTM's bidirectional gating for comprehensive contextual awareness, closely and smoothly tracks the true power fluctuations. Even under extreme conditions, the model effectively suppresses error amplification, demonstrating strong dynamic responsiveness and robustness to environmental variability.

Furthermore, to quantitatively assess the overall fitting performance of the proposed BiLSTM–Transformer hybrid model across the full output range, a scatter plot of predicted versus actual wind power for all test set samples was generated (as shown in Figure 3). In Figure 3, the horizontal axis represents the observed wind power, the vertical axis represents the model predictions, and the diagonal black dashed line corresponds to the ideal $y = x$ fit.

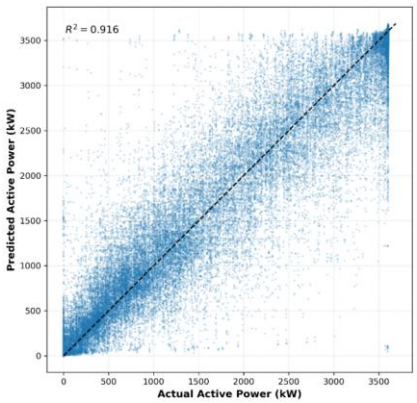

Figure 3: Scatter Plot of Predicted versus Actual Wind Power

It can be observed that a large number of sample points are tightly clustered around the ideal fit line, forming an elongated linear distribution without severe nonlinear distortion or systematic bias. Quantitative evaluation shows that the model achieves a coefficient of determination (R^2) of 0.916 on the test set, indicating that the predicted values account for the vast majority of the variance in the observed data. This further confirms that the proposed model not only excels in dynamic time series tracking but also demonstrates outstanding accuracy and robustness in overall regression fitting.

5. Conclusion

This study addresses the high volatility, strong randomness, and non-stationarity of wind power time series by proposing a BiLSTM–Transformer hybrid short-term forecasting model. The dual-channel parallel architecture extracts local temporal features and global long-range dependencies separately, while a dynamic fusion mechanism integrates them to achieve complementary advantages. Experiments on real-world wind farm SCADA data demonstrate that the proposed framework outperforms standalone baseline models across all key evaluation metrics. Specifically, the model achieves a Mean Absolute Error (MAE) of 181.24 kW, a Root Mean Square Error (RMSE) of 202.58 kW, a Mean Absolute Percentage Error (MAPE) of 6.16%, and a coefficient of determination (R^2) of 0.916. It also exhibits superior dynamic tracking capability and robustness under complex conditions, such as abrupt wind speed changes. This approach provides high-precision support for wind power grid integration and shows significant potential for further extension into spatiotemporal joint forecasting models incorporating Graph Neural Networks (GNNs).

References

[1] SUN Rongfu, ZHANG Tao, HE Qing, et al. Review on Key Technologies and Applications in Wind Power Forecasting [J]. High Voltage Engineering, 2021, 47(04): 1129-1143.
[2] HAN Zifeng, JING Qianming, ZHANG Yankai, et al. Review of wind power forecasting methods and new trends [J]. Power System Protection and Control, 2019, 47(24): 178-187.
[3] Li Junqing, Li Qiujia, Shi Tianyu, et al. Feature selection method for wind power prediction based on data mining [J]. Electrical Measurement & Instrumentation, 2019, 56(10): 87-92.
[4] QI Fangzhong, ZHUO Kexiang, ZHANG Jingya, et al. A short-term wind power prediction system and method based on multi-feature fusion and variational mode decomposition optimization algorithm [J]. Systems Engineering — Theory & Practice, 2025, 45(03): 1047-1064.
[5] FU Yang, Ren Zixu, WEI Shurong, et al. Ultra-short-term Power Prediction of Offshore Wind Power Based on Improved LSTM-TCN Model [J]. Proceedings of the CSEE, 2022, 42(12): 4292-4303.
[6] Dhaka P, Sreejeth M, Tripathi M M. Hybrid attention-based deep learning model using feature engineering approaches for wind power forecasting [J]. Electrical Engineering, 2025: 1-15.
[7] LUO Zhao, WU Yuhou, ZHU Jiaxiang, et al. Wind Power Forecasting Based on Multi-scale Time Series Block Auto-encoder Transformer Neural Network Model [J]. Power System Technology, 2023, 47(09): 3527-3537.
[8] ZHANG Yali, WANG Cong, ZHANG Hongli, et al. Multi-step Prediction of Ultra-short-term Wind Power Based on Non-stationary Transformer [J]. Smart Power, 2024, 52(01): 108-115.
[9] Pan X, Wang L, Wang Z, et al. Short-term wind speed forecasting based on spatial-temporal graph transformer networks [J]. Energy, 2022, 253.
[10] Li J, Li J, Li J, et al. Bayesian-Optimized GCN-BiLSTM-Adaboost Model for Power-Load Forecasting [J]. Electronics, 2025, 14(16): 3332.
[11] B. İşen. "Wind Turbine SCADA Dataset," Kaggle, 2018. [Online]. Available:https://www.kaggle.com/datasets/berkerisen/wind-turbine-scada-dataset.