Proceedings of the 3rd International Conference on Mechatronics and Smart Systems
DOI: 10.54254/2755-2721/141/2025.21570

Wind Power Prediction Based on LSTM and Self-Attention Mechanism

Yiheng Yang[1,a,*]

[1]*Harbin Institute of Technology, Shenzhen, HIT Campus of University Town of Shenzhen, Shenzhen, China*
a. 970024747@qq.com
**corresponding author*

Abstract: With the intensification of global climate change and energy crises, wind energy, as a clean and renewable energy source, has gradually become a crucial component in the energy sector. However, the intermittent and unstable nature of wind power generation poses significant challenges to accurately predicting the power output of wind turbines. This study proposes a wind power prediction model combining Long Short-Term Memory (LSTM) networks and Self-Attention mechanisms. LSTM net- works effectively capture long-term dependencies in time series through their gat- ing mechanisms, while the Self-Attention mechanism dynamically adjusts attention to critical time steps, further enhancing prediction accuracy. Experimental validation on real-world wind power datasets demonstrates that the LSTM + Attention model outperforms traditional RNN and LSTM models in terms of training loss, validation loss, and prediction accuracy, particularly in reducing prediction errors and improving accuracy. The results indicate that the LSTM model integrated with Self-Attention ef- fectively addresses complex nonlinear features in wind power prediction, enhancing both generalization capability and prediction precision. This model provides an ef- fective solution for wind power prediction and holds significant application value for optimizing grid dispatch and management, as well as improving the competitiveness of wind energy in the energy market.

Keywords: wind power generation, power prediction, long short-term memory (lstm), self-attention mechanism, deep learning

1. Introduction

The challenges of global climate change and energy crises have driven the growing demand for renewable energy. Among various renewable energy sources, wind energy has garnered significant attention due to its clean and sustainable characteristics. The reliability and stability of wind power generation largely depend on accurate power prediction capabilities. For grid operators, precise pre- diction of wind turbine output not only optimizes grid dispatch and management but also enhances the competitiveness of wind energy in the energy market. However, achieving high-precision power prediction remains a formidable challenge due to the intermittent and unstable nature of wind energy. In recent years, time series prediction techniques have demonstrated broad potential across various applications, such as financial market forecasting, weather prediction, and resource allocation. In the field of time series prediction, traditional models include Autoregressive (AR),

DOI: 10.54254/2755-2721/141/2025.21570

Autoregressive Moving Average (ARMA), and Autoregressive Integrated Moving Average (ARIMA).[1] However, these methods rely on linear assumptions and struggle to capture complex nonlinear features in time series data, limiting their practical applicability. To overcome these limitations, deep learning models have been introduced to time series prediction,[2] leveraging their powerful nonlinear mapping capabilities to more effectively identify implicit patterns and features in data.[3][4]

Wind power prediction methods have been extensively studied in recent years, leading to various approaches, including physical methods, statistical methods, and hybrid methods[5][6]. In wind power prediction, physical methods rely on Numerical Weather Prediction (NWP) data, combined with terrain characteristics and aerodynamic parameters, to estimate wind speed and calculate power output using wind turbine power curves[6]. For example, Focken et al. developed a physics-based wind power prediction method capable of forecasting wind power output up to 48 hours ahead[7]. How- ever, the high computational complexity of physical methods requires substantial resources, limiting their real-time application.

On the other hand, statistical methods analyze the relationship between historical wind speed and power data, with common approaches including time series analysis (e.g., ARIMA) and Artificial Neural Networks (ANN)[5]. For instance, the Wind Power Prediction Tool (WPPT) model, based on a nonlinear statistical model, provides short-term forecasts up to 36 hours[5]. Additionally, Firat et al. proposed a statistical method combining Independent Component Analysis (ICA) with Autoregressive (AR) models, achieving high accuracy in short-term predictions[8].

Hybrid methods combine the strengths of physical and statistical approaches to capture wind speed variations while improving prediction accuracy. For example, the Previento model integrates physical and statistical methods to deliver wind power forecasts up to 48 hours ahead[5]. Lin et al. employed a hybrid approach combining deep learning with Isolation Forest to detect data anomalies and enhance prediction accuracy, demonstrating the significant potential of hybrid methods in improving predictive capabilities[9].

In recent years, the emergence of deep learning techniques has highlighted the effectiveness of models such as Long Short-Term Memory (LSTM) networks in wind power prediction[10]. Zhang et al. utilized LSTM combined with a Gaussian Mixture Model (GMM) to achieve high accuracy in forecasting for a wind farm in northern China[11]. Furthermore, the ANEMOS project[5] integrates multiple prediction methods to provide wind power forecasts ranging from short-term to long-term, showcasing the feasibility of combining deep learning with traditional approaches.

In summary, this study proposes a model that integrates Self-Attention mechanisms with LSTM to fully leverage critical information in time series data, enabling more accurate prediction of wind turbine power output and providing feasible dispatch strategies and decision support.

2. Theoretical Foundations

2.1. Long Short-Term Memory Networks (LSTM)

Long Short-Term Memory (LSTM) networks are an enhanced variant of Recurrent Neural Net- works (RNNs), designed to address the challenge of learning long-term dependencies in traditional RNNs.[12][13] LSTM introduces memory cells and three gating mechanisms (input gate, forget gate, and output gate) to control information flow and storage, thereby effectively capturing both short- and long-term features in time series.[14] Traditional RNNs suffer from severe gradient vanishing issues in long-term dependency problems[15], whereas the gating mechanisms effectively mitigate gradient vanishing during sequence processing.[16]

Specifically, the memory cell serves as the main pathway for information transmission, retaining relevant information over extended periods. The input gate determines the extent to which current

DOI: 10.54254/2755-2721/141/2025.21570

input influences the memory cell, while the forget gate controls the degree of ”forgetting” past information.[17] The output gate regulates the contribution of the memory cell's current state to the output layer. This design allows LSTM to flexibly handle dependencies in time series without losing critical contextual information.[18] Through these gating structures, LSTM overcomes gradient vanishing or explosion issues, making it suitable for capturing complex nonlinear relationships across time steps in sequences.

Mathematically, for an input x_t at time step t and the previous hidden state h_{t-1}, the LSTM state update process is as follows:

- **Input gate**: Determines the influence of the current input on the memory cell.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

- **Forget gate**: Controls the update degree of the memory cell.[17]

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

- **Candidate memory cell state**: Generates a candidate state through the current input.

$$\tilde{C_t} = tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

- **Memory cell state update**: Integrates the effects of the forget gate and input gate.

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C_t}$$

- **Output gate**: Controls the output of the current hidden state.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t \cdot tanh(C_t)$$

Here, σ denotes the Sigmoid activation function, and W and b are model parameters. Through the memory cell state C_t and hidden state h_t, LSTM enables information to be retained or transmitted over extended periods within the network.

The gating mechanisms and state update process (illustrated in Figure 1) allow LSTM to effectively address long-term dependency issues and provide dynamic memory capabilities for time series prediction.

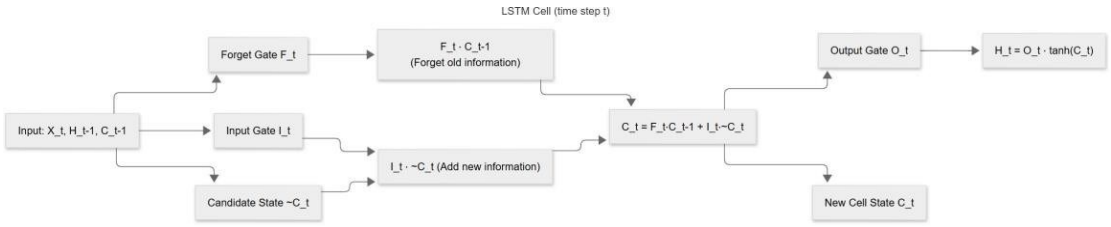

Figure 1: LSTM Cell Computational Flowchart

2.2. Self-Attention Mechanism

The Self-Attention mechanism is a structure designed to enhance a model's focus by dynamically adjusting attention to different positions in an input sequence. Unlike traditional sequence models, the Attention mechanism is not constrained by sequential order but calculates relevance based on matches between queries (Q), keys (K), and values (V) to identify the importance of each part of the sequence for the current task.[19]

DOI: 10.54254/2755-2721/141/2025.21570

In the Attention mechanism, each element of the input sequence generates query, key, and value vectors. Attention weights are computed via the dot product of queries and keys, followed by Softmax normalization.[20] The specific computation is as follows:

- **Attention weight calculation**: Compute attention weights for each input pair using queries Q and keys K.

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

Here, d_k represents the dimensionality of the key vectors, used to scale the dot product results to prevent gradient vanishing.

- **Weighted summation**: The weights multiplied by the value vectors V yield the Attention output, enabling the model to focus on critical information in the input sequence.

The core of the Attention mechanism lies in its ability to provide flexible contextual focus, allowing the network to dynamically adjust attention to different parts of the input sequence.[21] When combined with LSTM, the Attention mechanism adds selective focus to LSTM's hidden states, enabling LSTM to concentrate on the most critical historical information at the current time step, thereby mitigating its limitations in capturing long-term dependencies.[22]

3. Theoretical Framework of the LSTM-A Model

Combining Long Short-Term Memory (LSTM) with the Self-Attention mechanism leverages the strengths of both approaches: LSTM captures long- and short-term dependencies in time series, while Self-Attention dynamically focuses on critical information in the sequence. The integrated framework consists of the following two main components.

3.1. Feature Extraction Layer (LSTM Layer)

The primary role of the LSTM layer in this model is to extract temporal features from wind power time series data and provide rich contextual information for subsequent Self-Attention processing. Al- though LSTM inherently captures temporal dependencies through its gating mechanisms, the complex influences of multiple factors on wind power variations mean that LSTM's hidden state sequences alone may not fully capture all critical temporal features. Therefore, integrating LSTM with Self- Attention forms a powerful feature extraction and weighted focusing module to further improve pre- diction accuracy.

During the feature extraction phase, the LSTM layer processes the input time series data and generates hidden state sequences $H = \{h_1, h_2, \ldots, h_T\}$ at each time step through its memory cells and gating mechanisms. These hidden states represent "abstracted patterns" of wind power variations over time. However, since wind power at different time steps may be influenced by external factors (e.g., weather changes, sudden wind speed fluctuations), certain time steps in the hidden state sequences may have a greater impact on current predictions than others. Thus, relying solely on LSTM outputs cannot guarantee equal attention to all time steps, necessitating the introduction of Self-Attention to further weight these hidden states.

3.2. Attention Weighting Layer (Attention Layer)

The Attention mechanism assigns weights to each time step by computing inter-step correlations, dynamically adjusting the model's focus on key information in the sequence. The specific steps are as follows:

DOI: 10.54254/2755-2721/141/2025.21570

1. **Query, key, and value generation**: Linearly transform hidden states h_t into queries Q, keys K, and values V:

$$Q_t = W_Q \cdot h_t, \qquad K_t = W_K \cdot h_t, \qquad V_t = W_V \cdot h_t$$

where W_Q, W_K, W_V are trainable parameters.

2. **Attention weight calculation**: Compute attention weights via the dot product of queries and keys, followed by scaling and normalization:

$$\mathrm{a_{ij}} = softmax\left(\frac{Q_i K_j^T}{\sqrt{d_k}}\right)$$

Here, α_{ij} denotes the attention weight of time step i on time step j, and d_k is the dimensionality of the key vectors.

3. **Weighted summation**: Generate context vectors z_i by weighted summation of value vectors V using attention weights:

$$z_i = \sum_{j=1}^{T} a_{ij} V_j$$

The resulting context vectors $Z = \{z_1, z_2, \cdots, z_T\}$ reflect the importance of each time step in the sequence and encapsulate global contextual information.

This integrated framework enhances LSTM's modeling capability through the Attention mechanism, improving both long-term dependency capture and critical information identification in complex time series prediction tasks.[23]

Figure 2 illustrates the architecture of the LSTM-Attention model. The input sequence is processed by the LSTM layer to extract temporal features, followed by dynamic context vector computation via the Attention mechanism, ultimately yielding power prediction results.

Figure 2: LSTM-Attention Model Flowchart

4. Experimental Design and Results Analysis

4.1. Experimental Setup

To validate the effectiveness of the proposed model, experiments were conducted on real-world wind power datasets. The dataset includes features such as wind direction, temperature, humidity, air pressure, wind speed, and actual power output. Data were normalized and split into training, validation, and test sets in a 7:1:2 ratio. The experiments were implemented using the PyTorch framework on an NVIDIA GPU, with Mean Squared Error (MSE) as the loss function, Adam as the optimizer, and a learning rate of 0.001.

The following three models were compared:

- **Traditional RNN model (Baseline)**: A conventional RNN model for power prediction.
- **LSTM model**: An LSTM network for time series prediction.
- **LSTM + Attention model**: Combines LSTM with Self-Attention to enhance focus on critical time steps.

DOI: 10.54254/2755-2721/141/2025.21570

4.2. Results Analysis

4.2.1. Training and Validation Loss

Table 1: Loss Comparison Across Models

Model	Training Loss	Validation Loss
Traditional RNN	0.0221	0.0168
LSTM	0.0224	0.0165
LSTM + Attention	0.0186	0.0138

Table 1 compares the training and validation losses across models. The results reveal significant differences in performance. In terms of training loss, the traditional RNN and LSTM models exhibit comparable values (0.022), indicating their ability to quickly learn basic data patterns. However, the LSTM model achieves slightly better validation loss than the RNN, demonstrating superior temporal feature capture. In contrast, the LSTM + Attention model achieves significantly lower training and validation losses (0.0186 and 0.0138, respectively), indicating that the Self-Attention mechanism enables the model to focus more effectively on critical time steps, thereby improving learning efficiency and generalization.

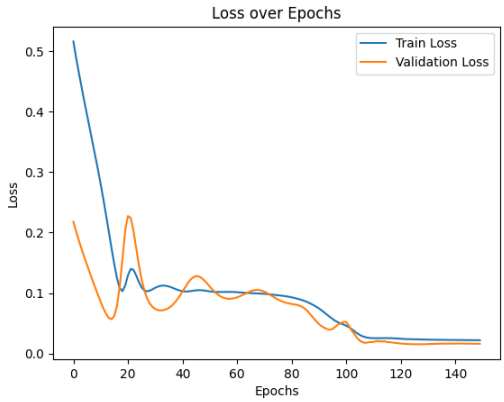

Figure 3: Training and Validation Loss of the Traditional RNN Model

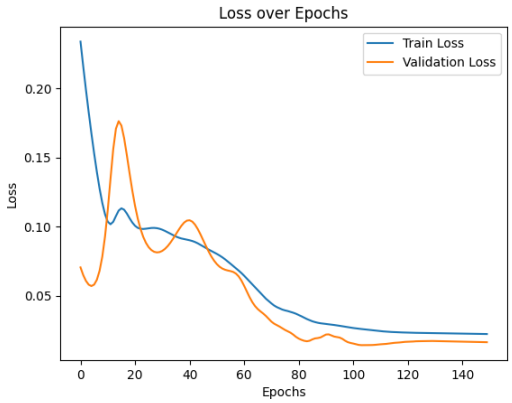

Figure 4: Training and Validation Loss of the LSTM Model

Figure 3, Figure 4, and Figure 5 respectively present the changes in training and validation losses of the traditional RNN model, LSTM model, and LSTM + Attention model across the number of training epochs.

DOI: 10.54254/2755-2721/141/2025.21570

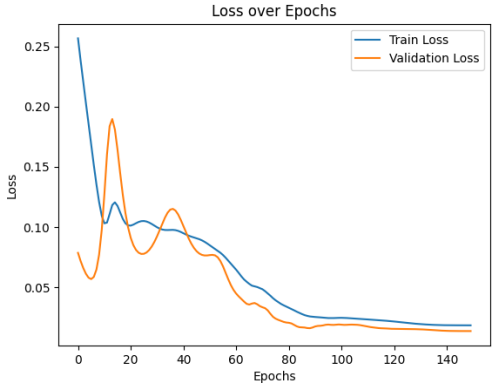

Figure 5: Training and Validation Loss of the LSTM + Attention Model

As can be observed from the figures, the traditional RNN model shows a rapid decrease in loss at the initial stage of training but tends to stabilize at a certain point, indicating that while the model can quickly learn the fundamental patterns of the data in the early stages, its performance is limited in more complex patterns. Compared to the traditional RNN, the LSTM model demonstrates better convergence in both training loss and validation loss, with a smaller validation loss, indicating a stronger capability for modeling time series. Meanwhile, the LSTM + Attention model exhibits a faster decline in loss during training and ultimately achieves the lowest validation loss, suggesting that the attention mechanism effectively enhances the model's focus on critical time steps, thus improving its generalization ability.

4.2.2. Prediction Performance Comparison

Through evaluations on the test set, the Root Mean Squared Error (RMSE) is used as the performance metric, and the results are shown in Table 2.

Table 2: Test RMSE for Different Models

Model	RMSE
Traditional RNN Model	0.3858
LSTM Model	0.3851
LSTM + Attention	0.3778

The results show that, compared with the traditional RNN model, the LSTM model can better capture the nonlinear features in time series data and reduce prediction errors. Building on this, the introduction of the Self-Attention mechanism in the LSTM + Attention model further improves prediction accuracy, reducing the RMSE to 0.3778, indicating that this model can more effectively focus on critical time steps and enhance prediction performance.

5. Conclusion

This study proposes a wind power prediction model that combines Long Short-Term Memory (LSTM) networks with the Self-Attention mechanism. By incorporating the Self-Attention mechanism, the model can dynamically focus on key time steps in the time series, effectively improving the accuracy of wind power prediction. Experimental results show that the LSTM + Attention model out- performs traditional RNN and LSTM models across multiple metrics, particularly in validation loss and Root Mean Squared Error (RMSE). Specifically, the LSTM + Attention model can enhance focus on crucial moments through the attention mechanism, further

DOI: 10.54254/2755-2721/141/2025.21570

improving the model's capability to model complex time series data and boosting prediction accuracy.

Compared with traditional RNN and LSTM models, the LSTM + Attention model shows a more rapid decline in loss during training and exhibits superior RMSE performance on the test set. Experiments demonstrate that integrating the Self-Attention mechanism with the LSTM model can more accurately capture the temporal characteristics of wind power, reduce prediction errors, and exhibit stronger adaptability and generalization capability, especially when dealing with the complex nonlinear features inherent in wind power data.

In summary, the combination of LSTM and the Self-Attention mechanism provides an effective solution for wind power prediction, enabling more accurate power forecasts in the wind power sector, optimizing power grid dispatch and management, and enhancing the potential for wind energy in the energy market. Future research could further explore the application of this model to other time series prediction tasks and improve prediction accuracy by integrating additional external information (such as weather forecasts).

References

[1] *Bri-Mathias Hodge, Austin Zeiler, Duncan Brooks, Gary Blau, Joseph Pekny, and Gintaras Reklatis. Improved wind power forecasting with arima models. In Computer aided chemical engineering, volume 29, pages 1789–1793. Elsevier, 2011.*

[2] *Guoqiang Zhang, B Eddy Patuwo, and Michael Y Hu. Forecasting with artificial neural net- works:: The state of the art. International journal of forecasting, 14(1):35–62, 1998.*

[3] *Ian Goodfellow. Deep learning, volume 196. MIT press, 2016.*

[4] *Gregor Giebel and George Kariniotakis. Wind power forecasting—a review of the state of the art. Renewable energy forecasting, pages 59–109, 2017.*

[5] *Xiaochen Wang, Peng Guo, and Xiaobin Huang. A review of wind power forecasting models. Energy Procedia, 12:770–778, 2011.*

[6] *Shahram Hanifi, Xiaolei Liu, Zi Lin, and Saeid Lotfian. A critical review of wind power fore- casting methods—past, present and future. Energies, 13(15):3764, 2020.*

[7] *Ulrich Focken, Matthias Lange, and Hans-Peter Waldl. Previento-a wind power prediction system with an innovative upscaling algorithm. In Proceedings of the European Wind Energy Confer- ence, Copenhagen, Denmark, volume 276, 2001.*

[8] *Umut Firat, Seref Naci Engin, Murat Saraclar, and Aysin Baytan Ertuzun. Wind speed forecasting based on second order blind identification and autoregressive model. In 2010 Ninth International Conference on Machine Learning and Applications, pages 686–691. IEEE, 2010.*

[9] *Zi Lin, Xiaolei Liu, and Maurizio Collu. Wind power prediction based on high-frequency scada data along with isolation forest and deep learning neural networks. International Journal of Electrical Power & Energy Systems, 118:105835, 2020.*

[10] *Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, and Gautam Shroff. Lstm-based encoder-decoder for multi-sensor anomaly detection. arXiv preprint arXiv:1607.00148, 2016.*

[11] *Jinhua Zhang, Jie Yan, David Infield, Yongqian Liu, and Fue-sang Lien. Short-term forecasting and uncertainty analysis of wind turbine power based on long short-term memory network and gaussian mixture model. Applied Energy, 241:229–244, 2019.*

[12] *S Hochreiter. Long short-term memory. Neural Computation MIT-Press, 1997.*

[13] *Kyunghyun Cho. On the properties of neural machine translation: Encoder-decoder approaches. arXiv preprint arXiv:1409.1259, 2014.*

[14] *Xin Liu, Luoxiao Yang, and Zijun Zhang. Short-term multi-step ahead wind power predictions based on a novel deep convolutional recurrent network method. IEEE Transactions on Sustain- able Energy, 12(3):1820–1833, 2021.*

[15] *Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. IEEE transactions on neural networks, 5(2):157–166, 1994.*

[16] *Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555, 2014.*

[17] *Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. Neural computation, 12(10):2451–2471, 2000.*

DOI: 10.54254/2755-2721/141/2025.21570

[18] I Sutskever. Sequence to sequence learning with neural networks. arXiv preprint arXiv:1409.3215, 2014.

[19] George Zerveas, Srideepika Jayaraman, Dhaval Patel, Anuradha Bhamidipaty, and Carsten Eick- hoff. A transformer-based framework for multivariate time series representation learning. In Pro- ceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining, pages 2114–2124, 2021.

[20] A Vaswani. Attention is all you need. Advances in Neural Information Processing Systems, 2017.

[21] Md Rasel Sarkar, Sreenatha G Anavatti, Tanmoy Dam, Mahardhika Pratama, and Berlian Al Kindhi. Enhancing wind power forecast precision via multi-head attention transformer: An investigation on single-step and multi-step forecasting. In 2023 International Joint Conference on Neural Networks (IJCNN), pages 1–8. IEEE, 2023.

[22] Hao Zhang, Jie Yan, Yongqian Liu, Yongqi Gao, Shuang Han, and Li Li. Multi-source and temporal attention network for probabilistic wind power prediction. IEEE Transactions on Sus- tainable Energy, 12(4):2205–2218, 2021.

[23] Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting, 37(4):1748–1764, 2021.