**Islamic University Journal of Applied Sciences (IUJAS)**
https://journals.iu.edu.sa/jesc

Volume VIII, Issue I, July 2026, Pages 01-21

# Comparative Analysis of Deep Recurrent Neural Network Models for Ultra Short-Term Wind Power Forecasting

Adel Mellit [1,2] *

[1] Faculty of Science and Technology, University of Jijel, 18000, Jijel, Algeria

[2] Abdus Salam International Center of Theoretical Physics (ICTP), Trieste, Italy

*Corresponding author: (Adel Mellit), *Email Address*: adel_mellit@univ-jijel.dz & a.mellit2@ictp.it

**Abstract**

The rapid growth of global wind power capacity has made accurate forecasting crucial for grid stability, economical allocation and efficient operation of wind farms. In this paper, we investigate the performance of various deep recurrent neural network (DRNN) models, including GRU, LSTM, BiGRU, and BiLSTM, deeper LTSM and LSTM-GRU for ultra short-term wind power forecasting. Furthermore, we incorporate a self-attention mechanism into LSTM (SA-LSTM) and GRU (SA-GRU) models to capture complex temporal dependencies and enhance forecasting accuracy. Extensive experiments on real-world wind power datasets demonstrate that the SA-LSTM and SA-GRU modes slightly outperform conventional architectures with an accuracy of 97%, while all evaluated models exhibit strong capabilities in capturing short-term fluctuations in wind power. These results emphasize the potential of advanced deep learning models to support real-time wind farm control, energy management and grid integration.

**Keywords:** Wind power; Wind speed; Forecasting; Deep Learning; Self-attention mechanism.

https://doi.org/10.63070/jesc.2026.014
Received 28 March 2026; Revised 24 April 2026; Accepted 28 April 2026.
Available online 02 May 2026.
Published by Islamic University of Madinah on behalf of *Islamic University Journal of Applied Sciences.*
This is an open access article published under the Creative Commons Attribution 4.0 International license (CC BY 4.0):https://creativecommons.org/licenses/by/4.0/

## 1. Introduction

Over the past decade, global wind power capacity has expanded dramatically, becoming one of the largest sources of renewable electricity generation. According to recent statistics, total installed wind power capacity worldwide has exceeded 1,100 gigawatts, with the vast majority coming from onshore installations and a growing share from offshore farms [1]. In 2024 alone, a record amount of new wind power capacity was added, reflecting the accelerated pace of deployment in key regions such as China, Europe, and the United States, and underscoring the pivotal role of wind power in reducing carbon emissions in the electricity sector and achieving global climate goals.

Ultra short-term wind forecasting, which typically covers time periods ranging from a few seconds to one hour (e.g., a 10-minute forecast), is critical for the real-time operation and stability of modern power systems, including microgrids [2]. Given the intermittent and rapidly changing nature of wind, accurate short-term forecasts enable system operators to maintain a balance between power generation and demand, ensuring grid frequency and voltage stability. Furthermore, these forecasts support effective participation in daily electricity markets by reducing imbalance costs and improving bidding strategies.

At the wind farm level, ultra short-term forecasting facilitates the implementation of advanced control strategies, including turbine tuning and power smoothing, which improves energy extraction and reduces mechanical stress [3]. Moreover, it plays a crucial role in improving energy storage management and anticipating sudden fluctuations in production, ultimately enabling greater reliance on renewable energy sources while reducing dependence on conventional backup power generation [4].

Over the past decade, various AI-based approaches have been developed to address the challenges of wind energy forecasting, with a primary focus on designing models that achieve the highest possible accuracy [5]. For example, in [6] a three-stage model to improve the accuracy of one-hour wind power forecasts was developed. The average absolute error was reduced from 0.333 to 0.278, with the SWD-ANFIS (Stationary Wavelet Decomposition Adaptive Neuro-fussy inference system) and (Artificial Neural Networks (ANN) models outperforming each other in the early stages. The ensemble predictions closely match actual data, though the reliance of the model on historical data and multiple modeling steps may limit real-time use. Another study [7] enhanced the National Center for Atmospheric Research system by combining numerical weather prediction and machine learning to improve short-term forecasts, quantify uncertainty, and predict extreme events such as icing. Quantile-

based data control and blended ramp forecasting improve accuracy, though the complexity of the system may limit real-time implementation.

In [8], the study showed that ANFIS provides the most accurate 1-hour-ahead wind power forecasts, with ANN and RBFN-OLS also performing well, while Radial Basis Function Network (RBFN)-Hybrid and RBFN-SGD underperform. Although all AI-based methods outperform persistence, the reliance on historical data and seasonal modeling may limit adaptability to sudden, unanticipated wind fluctuations. The proposed Self-Attention Temporal Convolutional Network (SATCN)-LSTM model in [9] effectively combines self-attention temporal convolution with Long short-term Memory (LSTM) to improve ultra-short-term wind power forecasting, outperforming LSTM, TCN, Convolutional Neural Network (CNN)-LSTM, and TCN-LSTM with up to 17.56% reduction in RMSE. While the authors in [10] demonstrated that the GRU model outperforms ARIMA in short-term wind power prediction, achieving high accuracy with root mean squared error (RMSE) of 0.047, mean absolute error (MAE) of 0.03, and a coefficient of 0.89. However, the effectiveness of the model depends on the quality and location-specific nature of the data, potentially limiting its generalization to other wind farms or regions.

In [11], a Bayesian bidirectional gated recurrent unit (BiGRU) model with encoder-decoder architecture and feature-temporal attention was proposed. But, the complexity of the model and dependance on advanced modeling and uncertainty quantification limit its real-time implementation and widespread adoption. To improve the forecasting accuracy another model based on dual-channel TCN-Informer architecture was proposed in [12]. The results proven that the model can effectively improves 15-minute and 1-hour wind power forecasts by decomposing sequences, classifying mode complexity, and applying specialized models to each group, achieving higher accuracy than traditional methods. Another work in [13] where the authors developed a proposed hybrid framework, by combining data analysis, model prediction and real-time error correction, which greatly improves the accuracy of very short-term wind energy prediction for both single-step and multi-step forecasts.

In [14], the study demonstrated that an AI-driven, preprocessing-enhanced framework using SCADA data significantly improves short-term wind power forecasting, with XGBoost achieving the highest accuracy ($R^2$=0.9890, RMSE=0.0385) by addressing nonlinearities and ensuring physically plausible predictions. The authors in [15] proposed a KSC-TFT framework, combining shape-based turbine clustering with Temporal Fusion Transformer (TFF) modeling, significantly improves ultra-short-term wind power forecasts by capturing turbine-level heterogeneity, reducing NMAE and NRMSE by over 50%, and enhancing probabilistic forecast accuracy [16]. It was shown in that FS-ISABO-VMD-IMSGNet framework significantly enhances ultra-short-term wind power forecasting, achieving an

accuracy of 28.2%. However, the multi-stage model design and complex optimization increase computational demands, which could limit its applicability in real-time operations.

In [17], the authors propose a combination forecasting method integrating Bidirectional LSTM (BiLSTM) with time attention and kernel density estimation to improve ultra short-term wind power predictions under extreme weather, achieving average improvements of 10.04% in PICP and 20.47% in PINAW. Another work in where the authors proposed a hybrid optimization of data, features, and model architecture, using DBSCAN for anomaly correction, correlation-based feature selection, and an improved informer with window attention and Fourier transform. The method achieves higher accuracy, better stability, and faster execution than state-of-the-art models, but its multi-stage process and reliance on complex preprocessing may increase computational demands, potentially limiting real-time deployment.

The main objective of this work is to investigate the performance of various deep learning (DL) models for ultra short-term wind power (WP) forecasting. A configuration is proposed that integrates a self-attention mechanism with LSTM and GRU networks (SA-LSTM and SA-GRU) to enhance forecasting accuracy. To assess the effectiveness of the proposed approach, a comprehensive comparative study is conducted among GRU, LSTM, BiGRU, BiLSTM, deeper LSTM, LSTM-GRU, SA-LSTM, and SA-GRU models, accompanied by a detailed analysis and discussion of their forecasting performance.

The remainder of this research is divided as follows: Section 2 presents the dataset used in developing the models, along with its analysis and preprocessing. Section 3 provides a detailed description of the proposed methodology, including the integration of attentional mechanisms with LSTM and GRU models. Section 4 presents the experimental results and evaluates the performance of the models studied, including both traditional DL-based iterative constructs and those enhanced with attentional mechanisms. Finally, the conclusions and future projections are presented in the last section.

## 2. Dataset analysis and processing

The dataset used in this study consists of wind power time series with a 10-minute forecasting horizon. The data were collected from the SCADA system of a wind turbine installed in southern Italy, spanning the period from 23/11/2005 to 28/12/2026, and comprising a total of 14,080 samples. As an example, Figure 1a illustrates the measured wind speed and the corresponding wind power generation.

The persistence of significant correlations across large lags indicates (See Fig.1b) non-stationary characteristics, which motivates the use of attention-based RNNs capable of capturing long-range dependencies.

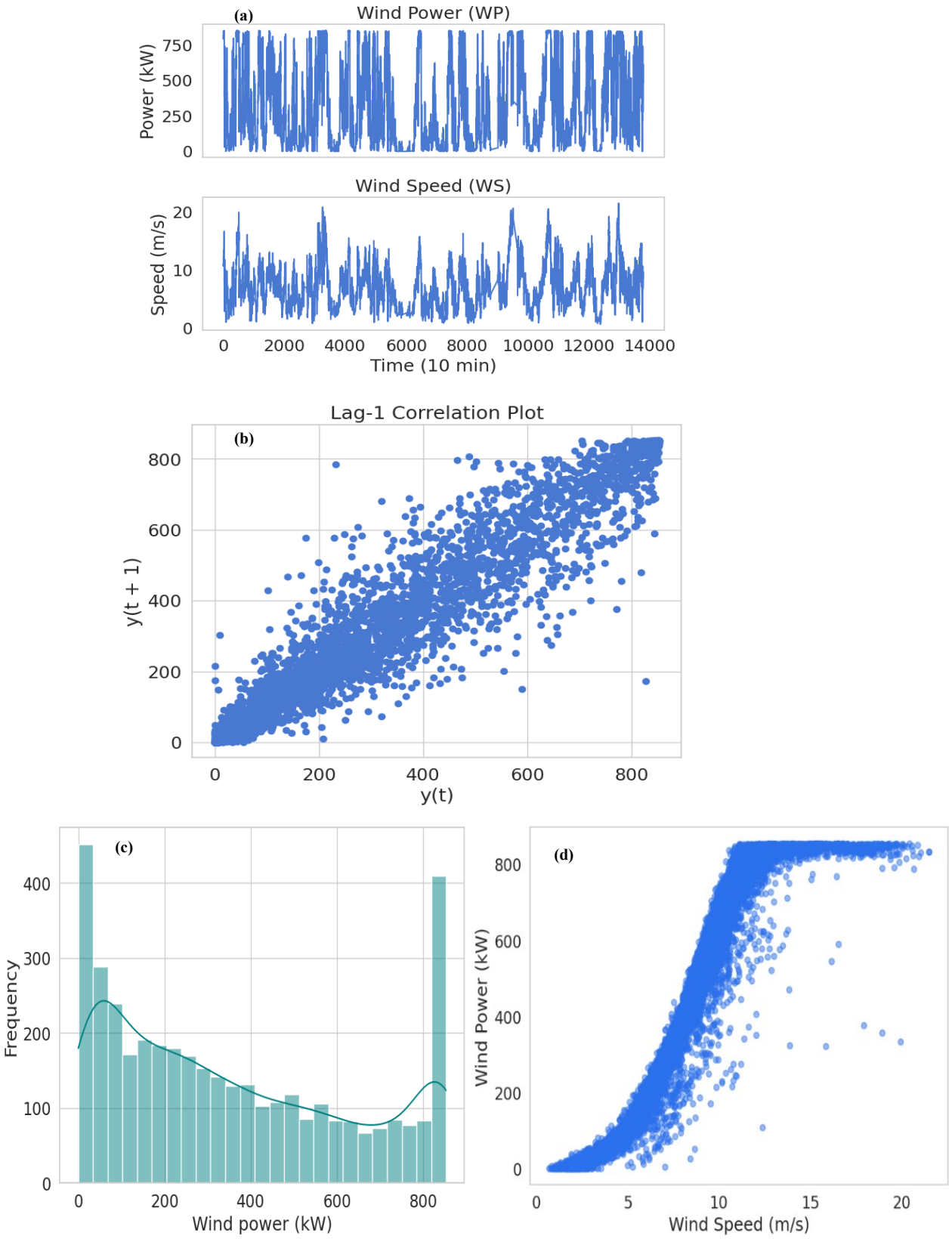



Fig. 1 a) Variation of wind power and wind speed over a period of 14000 samples, b) Lag-1 correlation plot ($y_{t+1}=f(y_t)$), c) Probability density function of WP variation and d) scatter curve of WP and WS.

The probability density function (See Fig.1c) for wind power exhibits a highly skewed multi-mode distribution, a characteristic feature of real-world wind power data. At low power levels (around 0-100 kW), a distinct peak is observed, indicating that the system often operates at low output. This behavior is generally consistent with low wind speed conditions or startup, where the turbine generates minimal power. As power increases (100-600 kW), the distribution gradually decreases, forming a long tail. This suggests that average power output occurs at a slower rate, reflecting the natural variability of wind speed over time. Interestingly, a secondary peak appears near the peak power range (around 800-850 kW). This indicates that the turbine often operates close to its rated power, most likely during periods of strong and stable wind conditions. This behavior is typically observed at high-efficiency wind farms. Figure 3d shows the scatter curve of WP and WS variation.

In this study, the historical WP data were preprocessed using Min-Max (See Eq.1) normalization to scale all values within the range [0, 1], ensuring that each feature contributes proportionally to the learning process and preventing dominance by larger values. The normalization is performed using the equation:

$$WP_{norm} = \frac{WP - WP_{min}}{WP_{max} - WP_{min}} \quad (1)$$

Where $WP_{norm}$ is the normalized value, $WP_{min}$ and $WP_{max}$ are minimum and maximum values of WP respectively. After normalization, the dataset was split into training and testing subsets, with 80% used to train the forecasting model and 20% reserved for testing and validation. Figure 2 illustrates the entire normalized wind power dataset along with the training and testing partitions. After removing outliers and negative values, the dataset size became 11,946. The first 80% of the data points (9,556), shown in blue, were used for model training, capturing the underlying temporal patterns of wind power generation. The remaining 20% (2390), shown in red color, was reserved for testing and validation to evaluate the predictive performance of the model on unseen data. This visualization emphasizes the temporal continuity of the dataset and the importance of maintaining separate datasets for unbiased model evaluation.

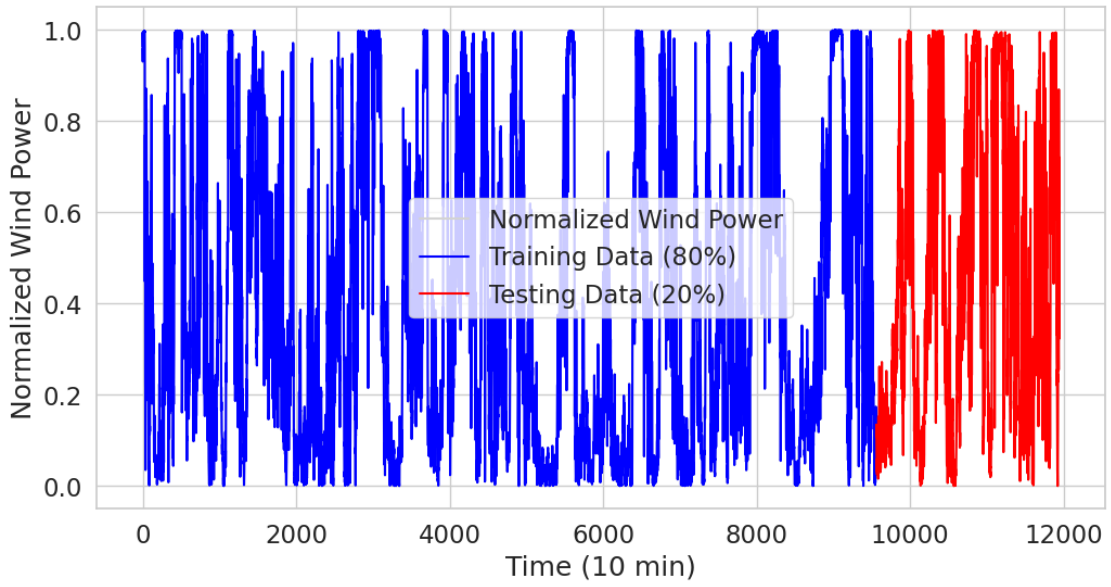



Fig. 2. Normalized WP dataset with 80% training and 20% testing split.

## 3. Methodology

### 3.1 Forecasting of wind power (WP)

Wind energy forecasting aims to predict future wind power values based on historical data. This process can be expressed mathematically as follows [19].

$$(\mathrm{WP}_{t+1},..,\mathrm{WP}_{t+k}) = f_{DNN}(\mathrm{WP}_{t-n},\mathrm{WP}_{t-n+1},\ldots,\mathrm{WP}_{t}) \quad (2)$$

$f_{\mathrm{DNN}}$ is a deep neural network (DNN), which will be used to estimate the future values of WP based on the fine-tuned parameters. In our case, $k$=1, so we will estimate one step ahead of WP (Next 10 min) and $t \in \{1, \ldots, n\}$, $n$ is the size of the sequence.

### 3.2 Deep RNN-based models forecasting of wind power

*3.2.1 Deep recurrent neural networks (DRNNs)*

The deep recurrent neural network (DRNNs) models examined include LSTM, GRU, BiGRU, BiLSTM, and LSTM-GRU models, as well as a deeper LSTM architecture. The predictive performance of these models depends heavily on the optimization of key hyperparameters, such as the number of layers, hidden units, and the length of the input sequence (search window), which controls

the amount of historical data used. Training parameters, including the learning rate, optimizer, batch size, and number of training cycles, affect convergence and stability. Furthermore, regulation techniques (such as dropout and L1/L2 penalties) are essential to prevent overshoot, particularly in deep and bidirectional models. Overall, fine-tuning of the hyperparameters is critical to ensuring the accuracy and reliability of very short-term wind energy forecasts.

*a) Long Short-Term Memory*

LSTM [20,21] is adopted in this study due to its effectiveness in modeling sequential data. Unlike conventional RNNs, LSTM introduces additional structural components, specifically, a memory cell and three gating mechanisms: input, output, and forget gates. These gates regulate the flow of information, enabling the network to selectively retain or discard information over time. This architecture effectively mitigates the vanishing gradient problem commonly encountered in standard RNNs, thereby allowing the model to capture long-term dependencies within time-series data. By combining one or more LSTM layers with a dense output layer, the resulting DRNN becomes particularly well-suited for time-series forecasting tasks, as illustrated in Fig. 3.

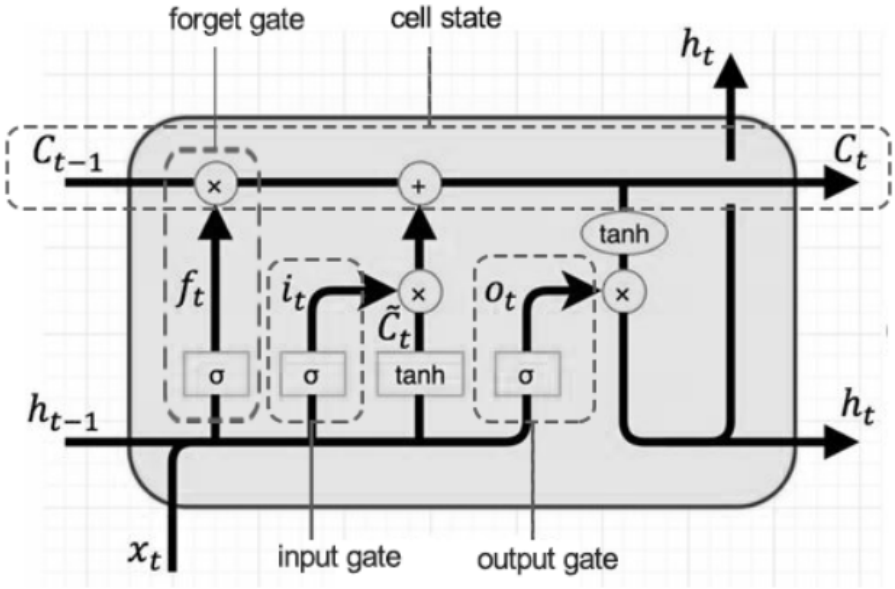



Fig. 3. Basic architecture of an LSTM deep neural network

Equations (3-8) mathematically describe the operation of the LSTM and how the information is processed from the input to the output passing through the respective gates:

$$i_t = \sigma\left(x_t U^i + h_{t-1} W^i\right) \quad (3)$$

$$f_t = \sigma(x_t U^f + h_{t-1} W^f) \quad (4)$$

$$o_t = \sigma(x_t U^0 + h_{t-1} W^0) \quad (5)$$

$$\hat{C}_t = tanh(x_t U^g + h_{t-1} W^g) \quad (6)$$

$$C_t = \sigma\left(f_t * C_{t-1} \quad + i_t * \hat{C}_t\right) \quad (7)$$

$$h_t = tanh(C_t) * o_t \quad (8)$$

Where $i_t$ is the input gate activation, $x_t$ is the current input, $h_{t-1}$ is the hidden state from previous step, $U_t$ is the weight matrix for the input xt Wt is the weight matrix for the previous hidden state $h_{t-1}$, σ is the sigmoid activation function, $f_t$ is the forget gate, $C_t$ is the cell candidate, $o_t$ is the output gate, $h_t$ is the hidden state.

*b) Gate recurrent unit*

The GRU [21,22] is a simplified variant of the LSTM architecture, designed to address the vanishing gradient problem encountered in training recurrent networks (see Fig. 4). Unlike LSTM, GRU employs two gating mechanisms—namely, the update gate and the reset gate, which regulate the flow of information within the network. These gates determine how much past information should be retained and how much new input should be incorporated into the current state. This streamlined structure reduces computational complexity while maintaining the ability to capture temporal dependencies. The mathematical formulation of the GRU is provided in Eqs. (9)–(12).

$$z_t = \sigma\left(x_t U^z + h_{t-1} W^z\right) \quad (9)$$

$$r_t = \sigma(x_t U^r + h_{t-1} W^r) \quad (10)$$

$$\hat{h}_t = tanh(x_t U^h + (r_t * h_{t-1}) W^h) \quad (11)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t \hat{h}_t \quad (12)$$

Here *r* is a reset gate, and *z* is an update gate

.

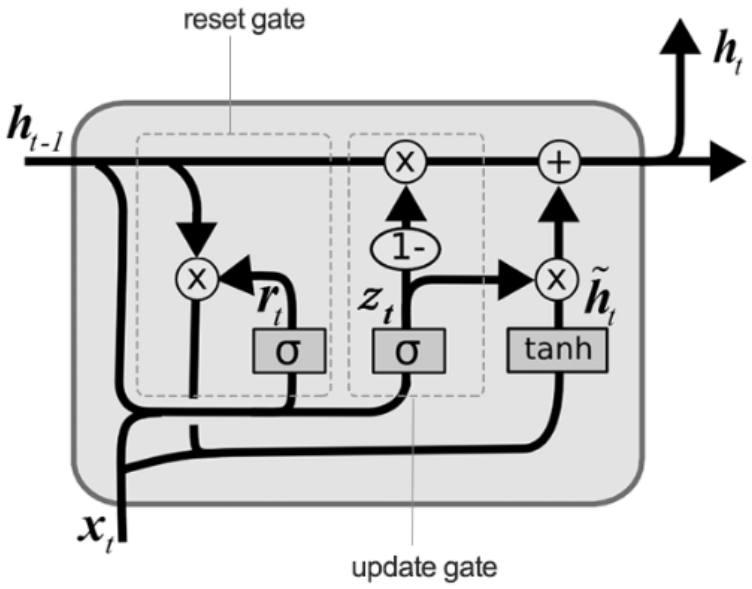



Fig. 4. Basic architecture of a GRU deep neural network.

*c)BiLSTM and BiGRU*

Bidirectional recurrent models, such as BiLSTM and BiGRU, extend conventional architectures by processing input sequences in both forward and backward directions (see Fig. 5) [23,24]. This dual processing enables the model to simultaneously capture dependencies from past (previous time steps) and future (subsequent time steps) contexts, leading to a more comprehensive representation of temporal dynamics. By combining the hidden states from both directionsm, typically through concatenation or summation, bidirectional models can better learn complex patterns and contextual relationships within time-series data. This capability often results in improved prediction accuracy, particularly in scenarios where the full sequence is available and future context provides valuable information.

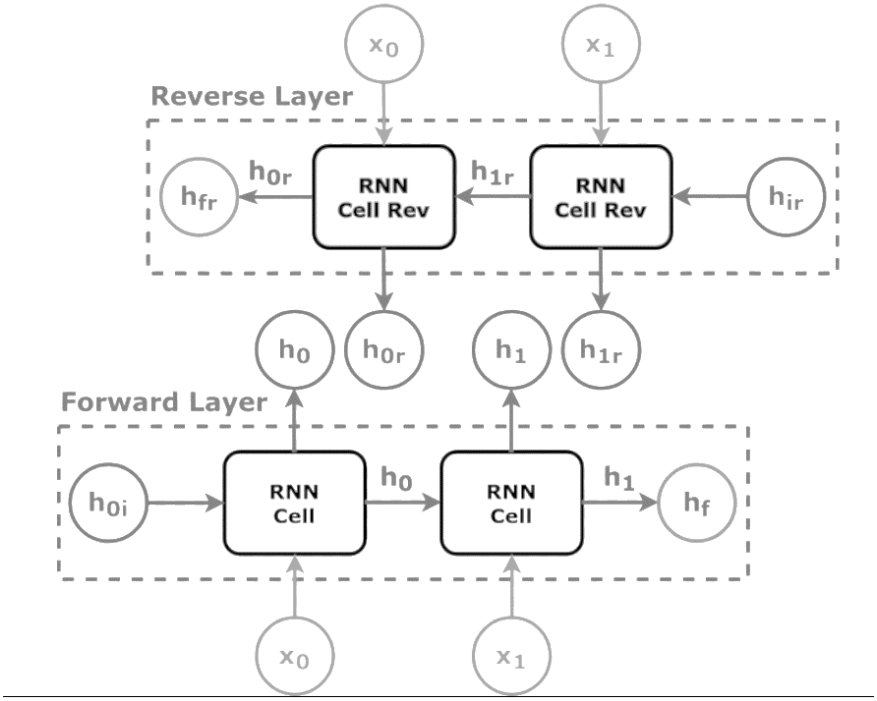



Fig. 5. Basic architecture Bidirectional RNN (BiGRU and BiLSTM).

They can increase the amount of information available to the network, improving the context available to the algorithm. The forward RNN processes in one direction:

$$f_\theta(x_0, h_0) = (y_0, h_1), f_\theta(x_1, h_1) = (y_1, h_2), \ldots \quad (13)$$

The backward RNN processes in the opposite direction:

$$f'_\theta(x_N, h'_0) = (y'_N, h'_{N-1}), f'_\theta(x_{N-1}, h'_{N-1}) = (y'_{N-1}, h'_{N-2}), \ldots \quad (14)$$

Where $f_\theta$ and $f'_\theta$ are the activation functions for forward and backword RNN.

*d) stacked and deeper RNNs (GRU and LSTM)*

Deep, stacked RNNs, including LSTM and GRU architectures, refer to models consisting of multiple sequentially ordered recurrent layers, where the output of each layer serves as the input for the next. This deep structure enables the network to learn hierarchical time representations, with lower layers capturing short-term patterns, while upper layers model more complex and abstract dependencies over longer timescales [26]. By increasing network depth, RNNs enhance the ability of the model to

represent nonlinear dynamics and improve prediction accuracy. However, deeper architectures introduce additional challenges, such as increased computational complexity and a higher risk of overshoot, necessitating fine-tuning of hyperparameters and the use of regulation techniques like dropout and early termination.

### *3.2.2 Self-attention mechanism in DRNNs*

Self-attention is an advanced mechanism that enhances recurrent neural networks (RNNs), including LSTM, GRU, BiLSTM, and BiGRU, by enabling them to dynamically focus on the most relevant parts of an input sequence when generating predictions. In conventional LSTM and GRU architectures, the hidden state at each time step summarizes past information; however, their ability to capture long-term dependencies may be limited due to the progressive attenuation of earlier information.

To overcome this limitation, self-attention computes a context vector as a weighted sum of all hidden states in the sequence, allowing the model to directly access both recent and distant information. The attention mechanism can be formulated as follows [27]:

$$e_t = v^T tanh(W_h h_t + b) \tag{15}$$

$$\alpha_t = \frac{exp(e_t)}{\sum_{k=1}^{T} exp(e_k)} \tag{16}$$

$$c = \sum_{t=1}^{T} \alpha_t \, h_t \tag{17}$$

where $e_t$ is the attention score, $\alpha_t$ denotes the normalized attention weight obtained via the Softmax function, and c is the resulting context vector.

This mechanism allows the model to assign higher importance to the most informative time steps while reducing the influence of less relevant ones, thereby improving the modeling of long-term dependencies [28]. Furthermore, when combined with bidirectional architectures such as BiLSTM and BiGRU, self-attention becomes even more effective by exploiting both past and future contextual information.

Consequently, self-attention significantly enhances the model's ability to capture complex temporal patterns and salient features, leading to improved performance in time-series forecasting, classification, and sequence generation tasks [28]. As an illustrative example, the following outlines the key steps for integrating self-attention into an LSTM (or GRU) model for time-series forecasting.

As an illustrative example, the following outlines the key steps for applying self-attention to an LSTM (or GRU) model in time series forecasting.

**Step #1:** Dataset

Given a time series:

$$X=\{x_1, x_2, \ldots x_K\} \tag{18}$$

Create sliding windows of length *n* as input and the next value as target:

$$X_{input}=[x_{t-n}, \ldots x_{t-1}],\ y_t=x_t \tag{19}$$

**Step #2:** LSTM layer or GRU

For each timestep *t*, LSTM computes hidden state $h_t$ and cell state $c_t$:

$$f_t = \alpha\left(W_f.[h_{t-1}, x_t] + b_f\right) \tag{20}$$

$$i_t = \alpha(W_i.[h_{t-1}, x_t] + b_i) \tag{21}$$

$$o_t = \alpha(W_o.[h_{t-1}, x_t] + b_o) \tag{22}$$

$$C_t = f_t C_{t-1} + i_t.th(W_c.[h_{t-1}, x_t] + b_c) \tag{23}$$

$$h_t = O_t.\beta(C_t) \tag{24}$$

Where σ is the sigmoid function, $x_t$ is the input, $f_t$ is the forget gate, $i_t$ is the input gate, $o_t$ is the output gate, $h_t$ is the output, $c_t$ is the cell state, $c_{t-1}$ is the cell update, β, α are the activation functions, $W_o$, $W_i$, $W_f$ and $W_c$ are the weights, and $b_f$, $b_i$, $b_o$ and $b_c$ are the bias.

**Step #3:** Compute attention weights

For a sequence of LSTM hidden states $H=[h_1,h_2,\ldots,h_n]$ and the current step hidden state $h_t$, compute attention score $e_{t,i}$ for each timestep *i*:

$$e_{t,i} = h_t W_a h_i \tag{25}$$

Normalize scores with SoftMax to get attention weights $\alpha_{t,i}$:

$$\alpha_{t,i} = \frac{exp(e_{t,i})}{\sum_{j=1}^{n} exp(e_{t,j})} \tag{26}$$

**Step #4:** Compute the context vector and combine it with the hidden state.

Weighted sum of all hidden states:

$$C_t = \sum_{j=1}^{n} \alpha_{t,i} h_i \tag{27}$$

Concatenate context vector with current hidden state:

$$\tilde{h}_t = tanh\left(W_c\left[\alpha_{t,i}; h_i\right]\right) \tag{28}$$

**Step #5:** Output prediction and update weights

Feed $h_t$ to a dense layer for forecasting

$$\hat{y}_t = W_0\tilde{h}_t + b_0 \quad (29)$$

Loss function (e.g., MSE)

$$MSE = \frac{1}{N}\sum_{t=1}^{N}(y_t - \hat{y}_t)^2 \quad (30)$$

*3.3.3 Fine-tuned parameters*

All the iterative models developed in this study, including LSTM, GRU, BiLSTM, BiGRU, deeper and stacked RNNs and attention mechanism-based variants (SA-LSTM and SA-GRU), were finely tuned to improve prediction accuracy. Key hyperparameters, such as hidden units, number of layers, sequence length, dropout rates, learning rate, batch size, and attention mechanism size, were optimized [29,30]. Early stop was implemented to prevent overshoot, while attention mechanisms were tuned to focus on the most critical time steps. This tuning ensured an effective balance between temporal dependency incorporation, generalization, and computational efficiency.

## 4. Results and discussion

To evaluate the performance of the investigated RNN-based models (LSTM, GRU, BiLSTM, and BiGRU), including their attention-enhanced variants SA-LSTM and SA-GRU, the error metrics shown in Appendix are used: Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), the coefficient of determination ($R^2$) (See the Appendix section).

Figure 6 a illustrates the evolution of the loss function (MSE), showing that a low final value of 0.003 was achieved. As an illustrative example, Figure 6 b presents a comparison between the measured and forecasted WP obtained using the developed SA-GRU model. A strong agreement between the measured and forecasted values can be observed. Furthermore, Figure 6c depicts the scatter plots, revealing a high correlation between the measured and predicted data, with an $R^2$ of 97.17 %.

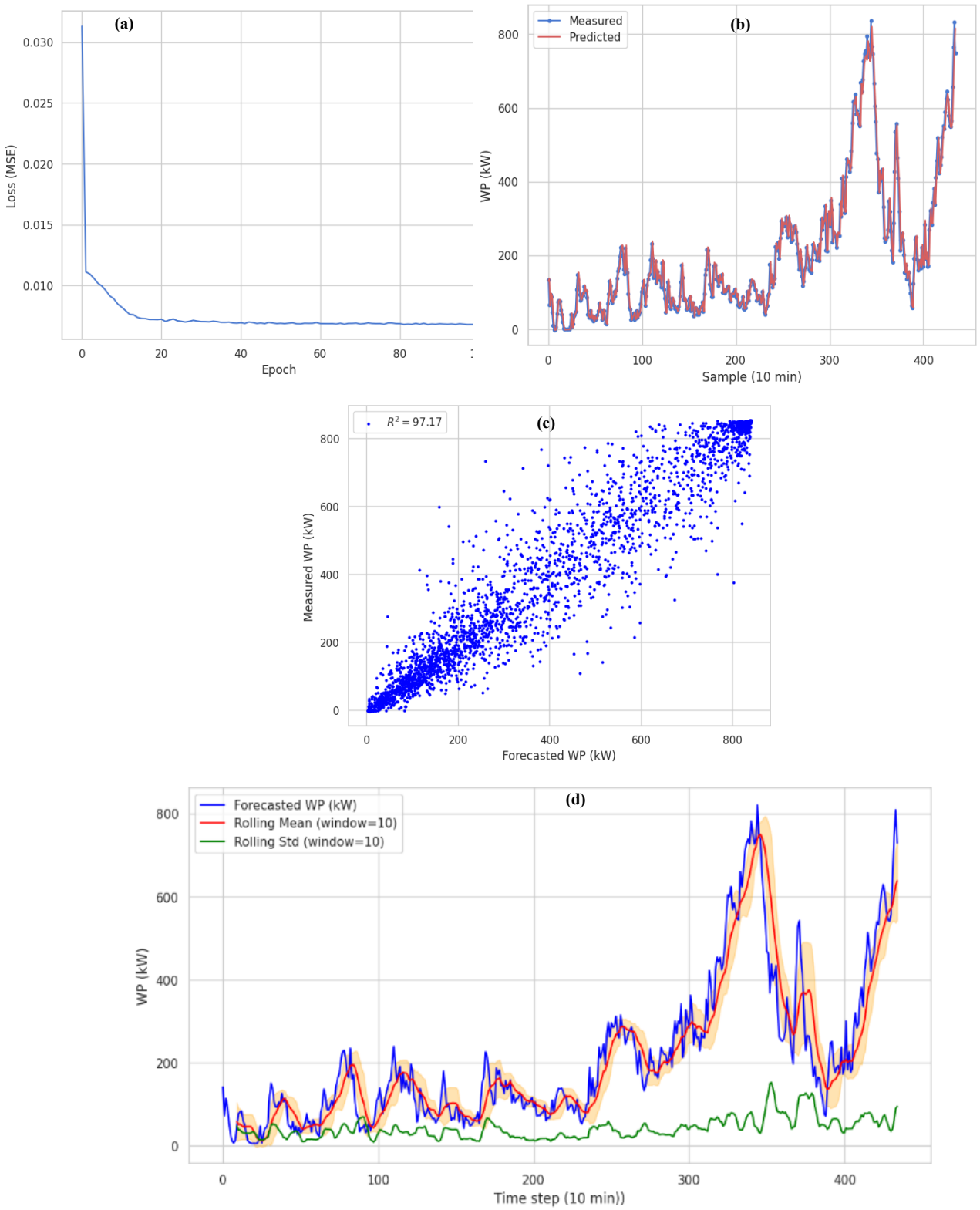



Fig. 6 a) Loss function, b) Measured vs. forecasted PW for a period of 450 samples, c) Scatter plot between measured and forecasted PW, and d) Forecasted PW with rolling statistics.

Figure 6 d illustrates the predicted WP over 450 time steps along with its rolling statistics. The blue curve represents the raw forecasted WP, showing high variability with sharp peaks and valleys characteristic of wind intermittency. The red curve depicts the rolling mean (window=10), which smooths short-term fluctuations and highlights the underlying trend in wind power generation. The green curve corresponds to the rolling standard deviation over the same window, indicating local volatility; higher values coincide with periods of rapid changes in WP, while lower values reflect more stable behavior. Together, the rolling mean and standard deviation provide a clear visual representation of both the trend and variability in the predicted WP, enabling the identification of stable periods versus highly volatile intervals. This approach is useful for assessing forecast reliability and for guiding further model evaluation or smoothing strategies..

Table 1 summarizes the performance of the developed models for short-term wind power forecasting using RMSE, MAPE, and $R^2$ as evaluation metrics. Basic models, including GRU, LSTM, and Bi-GRU, achieved similar RMSE values (0.091-0.093 kW), although BiGRU exhibited a high MAPE (56.74%), indicating larger relative errors at lower power levels. Deeper and hybrid architectures improved accuracy; for instance, the Deeper LSTM reduced RMSE to 0.080 kW and MAPE to 22.54%.

Table 1. Comparison of the developed models for the PW forecasting.

| **Model** | **RMSE (kW)** | **MAPE (%)** | **$R^2$ (%)** |
|---|---|---|---|
| **GRU** | 0.093 | 28.77 | 96.14 |
| **LSTM** | 0.092 | 25.03 | 96.17 |
| **BiGRU** | 0.091 | 56.74 | 96.38 |
| **BiLSTM** | 0.093 | 34.78 | 96.25 |
| **LSTM-GRU** | 0.092 | 30.28 | 96.10 |
| **Deeper LSTM** | 0.080 | 22.54 | 96.51 |
| **SA-LSTM** | 0.074 | 15.90 | 97.20 |
| **SA-GRU** | 0.075 | 15.87 | 97.13 |

Models incorporating attention mechanisms, such as SA-LSTM and SA-GRU, achieved the best performance, with RMSE between 0.074 and 0.075 kW, MAPE around 15.9%, and $R^2$ approximately 0.97, demonstrating their effectiveness in capturing the most relevant temporal patterns. Figure 7a compares RMSE, MAPE, and $R^2$ across all models, while Figure 7b presents a composite ranking highlighting the superior performance of attention-based frameworks.

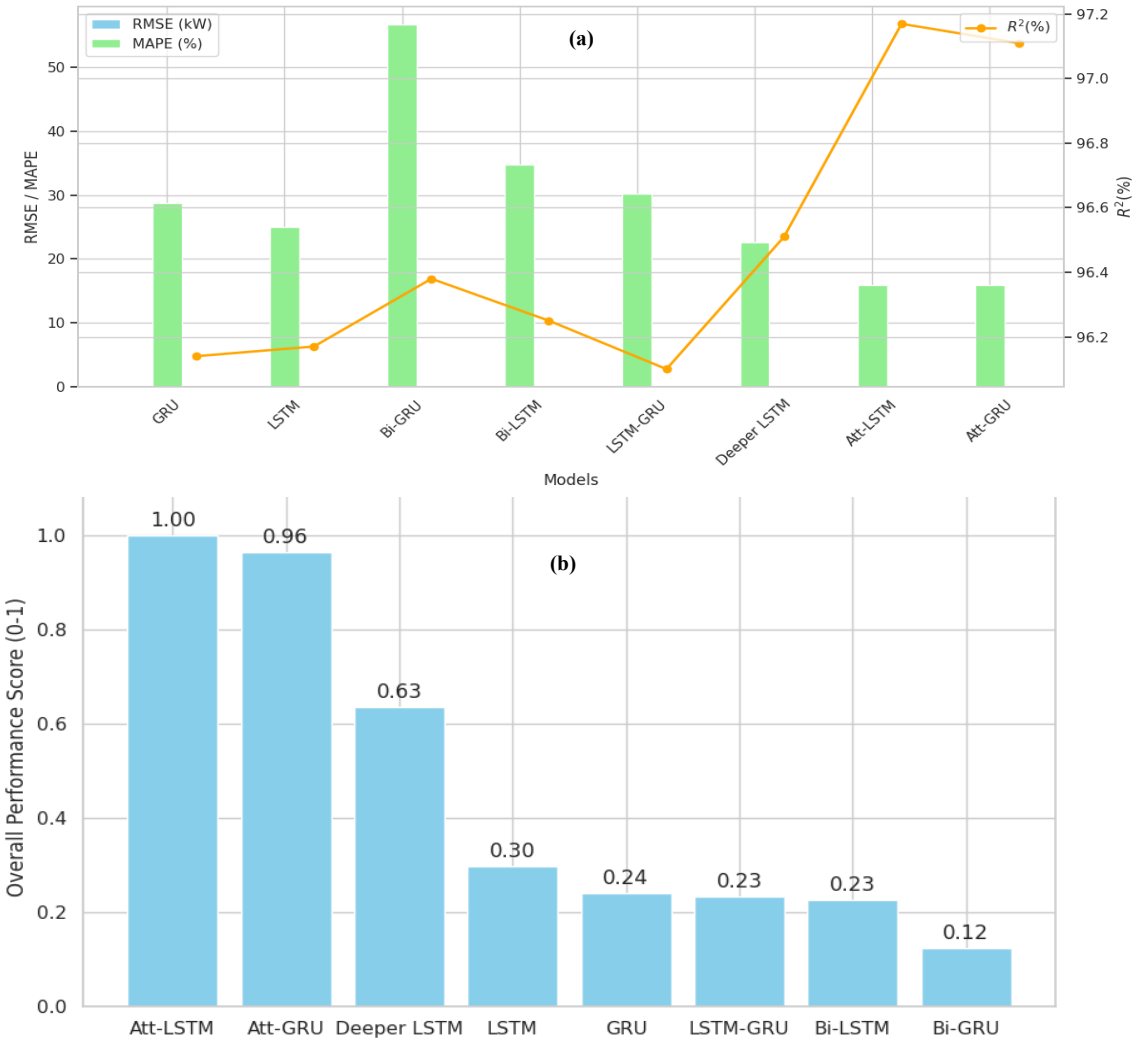



Fig. 7. a) Comparison of WP forecasting models and b) Ranking of WP forecasting models based on composite Score

## 5. Conclusion and perspectives

In this work, we assessed several DRNN models for ultra short-term wind power forecasting. The findings highlight the capability of these models to provide highly accurate wind power predictions. The forecasted wind power can be efficiently used for microgrid control and energy management. Among the evaluated models, the developed GRU and LSTM architectures, based on an attention mechanism (SA-GRU and SA-LSTM), showed clear advantages over standard deep RNN

architectures. They were able to better detect time dependencies, focus on the most relevant historical patterns, and reduce both RMSE and the MAPE, thus improving overall predictive performance.

Future research directions include scaling these models to multi-site and geographically distributed wind farms, incorporating probabilistic forecasting to quantify uncertainty, and exploring lightweight DL architectures suitable for immediate deployment in embedded systems. Furthermore, integrating weather forecasts and other environmental variables may further enhance the accuracy and robustness of the forecasts.

## Appendix

*Error metrics*

RMSE measures the square root of the average squared difference between the predicted and measured values, providing an indication of the overall prediction accuracy while penalizing large errors more heavily

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{M}\left(WP_i - \widehat{WP}_i\right)^2}$$

where $WP_i$ and $\widehat{WP}_i$ represent the measured and predicted values, respectively, and N is the number of samples.

MAPE quantifies the average relative error between predicted and actual values in percentage form, making it particularly useful for assessing forecasting accuracy in a normalized manner. It is expressed as:

$$MAPE = \sqrt{\frac{100}{M}\sum_{i=1}^{M}\frac{(WP_i - \widehat{WP}_i)}{WP_i}}$$

$R^2$ evaluates the strength of the linear relationship between predicted and measured values, indicating how well the model captures the temporal variability of the data. It is given by:

$$R^2 = \frac{\sum_{i=1}^{M}(WP_i - \overline{WP}_i)\left(\widehat{WP}_i - \overline{\widehat{WP}_i}\right)}{\sum_{i=1}^{M}(WP_i - \overline{WP}_i)^2 \sum_{i=1}^{N}\left(\widehat{WP}_i - \overline{\widehat{WP}_i}\right)^2}$$

where $\overline{WPi}$ and $\overline{\widehat{WPi}}$ denote the mean values of the measured and predicted data, respectively. Together, these metrics provide a comprehensive assessment of model accuracy, relative error magnitude, and correlation with observed data.

**Acknowledgements**

The author would like to acknowledge support from the International Center of Theoretical Physics (ICTP), Trieste (Italy) through the Associates Program (2023–2028).

**Funding**

The authors declare that this research received no external funding.

**Data Availability Statement**

Not applicable.

**Conflicts of Interest**

There is no conflict of interest, the writers assert.

**References**

[1] Wind power capacity in the Net Zero Scenario, 2015–2030, Int. Energy Agency (IEA). Available: https://www.iea.org/data-and-statistics/charts/wind-power-capacity-in-the-net-zero-scenario-2015-2030

[2] F. Rodríguez, A.M. Florez-Tapia, L. Fontán, A. Galarza, Very short-term wind power density forecasting through artificial neural networks for microgrid control, Renew. Energy 145 (2020) 1517–1527.

[3] G. Yu, L. Shen, Q. Dong, G. Cui, S. Wang, D. Xin, X. Chen, W. Lu, Ultra-short-term wind power forecasting techniques: Comparative analysis and future trends, Front. Energy Res. 11 (2024) 1345004.

[4] W.C. Tsai, C.M. Hong, C.S. Tu, W.M. Lin, C.H. Chen, A review of modern wind power generation forecasting technologies, Sustainability 15 (2023) 10757.

[5] M.H. Lipu, M.S. Miah, M.A. Hannan, A. Hussain, M.R. Sarker, A. Ayob, M.H.M. Saad, M.S. Mahmud, Artificial intelligence based hybrid forecasting approaches for wind power generation: Progress, challenges and prospects, IEEE Access 9 (2021) 102460–102489.

[6] H.H. Çevik, M. Çunkaş, K. Polat, A new multistage short-term wind power forecast model using decomposition and artificial intelligence methods, Physica A 534 (2019) 122177.

[7] B. Kosovic, S.E. Haupt, D. Adriaansen, S. Alessandrini, G. Wiener, L. Delle Monache, Y. Liu, S. Linden, T. Jensen, W. Cheng, M. Politovich, A comprehensive wind power forecasting system integrating artificial intelligence and numerical weather prediction, Energies 13 (2020) 1372.

[8] M. Godinho, R. Castro, Comparative performance of AI methods for wind power forecast in Portugal, Wind Energy 24 (2021) 39–53.

[9] L. Xiang, J. Liu, X. Yang, A. Hu, H. Su, Ultra-short term wind power prediction applying a novel model named SATCN-LSTM, Energy Convers. Manag. 252 (2022) 115036.

[10] S. Qureshi, F. Shaikh, L. Kumar, F. Ali, M. Awais, A.E. Gürel, Short-term forecasting of wind power generation using artificial intelligence, Environ. Chall. 11 (2023) 100722.

[11] L. Liu, J. Liu, Y. Ye, H. Liu, K. Chen, D. Li, X. Dong, M. Sun, Ultra-short-term wind power forecasting based on deep Bayesian model with uncertainty, Renew. Energy 205 (2023) 598–607.

[12] Q. Li, X. Ren, F. Zhang, L. Gao, B. Hao, A novel ultra-short-term wind power forecasting method based on TCN and Informer models, Comput. Electr. Eng. 120 (2024) 109632.

[13] S. Yang, J. Zhou, X. Gu, Y. Mei, J. Duan, A comprehensive framework of the decomposition-based hybrid method for ultra-short-term wind power forecasting with on-site application, Energy 313 (2024) 133911.

[14] B. Kishore, K. Kabilan, L.S. Rahul, G.P. Priyadarshan, E.P. Vishnutheerth, R. Satheesh, M.L. Kolhe, Advancing short-term wind power forecasting by AI-driven models for improved accuracy, Electr. Eng. 107 (2025) 14795–14810.

[15] C. Zhang, X. Xiao, Y. Wang, M.Z. Hou, S. Huang, W. Hu, M. Hu, R. Huang, Fine-grained ultra-short-term wind power forecasting based on Temporal Fusion Transformers integrated with turbine power time series clustering, Energy (2025) 137995.

[16] X. Shu, L. Gao, F. Zhang, X. Ren, L. Qin, Y. Wang, X. Wu, Ultra-short-term wind power forecasting method based on optimized decomposition and deep learning, Energy Convers. Manag. X (2025) 101315.

[17] W. Li, T. Yang, J. Yang, L. Peng, Predicting ultra-short-term wind power combinations under extreme weather conditions, IEEE Access (2025).

[18] P. Xu, S. Zhang, P. Luo, X. Wang, Y. Yan, D. Qi, Data-feature hybrid optimization for ultra-short-term wind power forecasting, Int. J. Electr. Power Energy Syst. 176 (2026) 111742.

[19] Mellit, A.M. Pavan, V. Lughi, Deep learning neural networks for short-term photovoltaic power forecasting, Renew. Energy 172 (2021) 276–288.

[20] Hochreiter, S. and Schmidhuber, J.,. Long short-term memory. Neural computation, 9(8), (1997) pp.1735-1780.

[21] Han, Z., Zhao, J., Leung, H., Ma, K.F. and Wang, W., 2019. A review of deep learning models for time series prediction. IEEE Sensors Journal, 21(6), pp.7833-7848.

[22] Boubaker, S., Benghanem, M., Mellit, A., Lefza, A., Kahouli, O. and Kolsi, L., Deep neural networks for predicting solar radiation at Hail Region, Saudi Arabia. IEEE Access, 2021, 36719-36729.

[23] Gamboa, J.C.B. Deep learning for time-series analysis. arXiv, 2017, preprint arXiv:1701.01887.

[24] Cho, K., Van Merriënboer, B., Bahdanau, D., & Bengio, Y. On the properties of neural machine translation: Encoder–decoder approaches. 2014, ArXiv:1409.1259.

[25] Xu, L., Xu, W., Cui, Q., Li, M., Luo, B. and Tang, Y., 2023. Deep heuristic evolutionary regression model based on the fusion of BiGRU and BiLSTM. Cognitive Computation, 15(5), pp.1672-1686.

[26] Sagheer, A. and Kotb, M.,. Unsupervised pre-training of a deep LSTM-based stacked autoencoder for multivariate time series forecasting problems. Scientific reports, 9(1), (2019), p.19038.

[27] W. Li, F. Qi, M. Tang, Z. Yu, Bidirectional LSTM with self-attention mechanism and multi-channel features for sentiment classification, Neurocomputing 387 (2020) 63–77.

[28] Kilinc, H.C., Apak, S., Ozkan, F., Ergin, M.E. and Yurtsever, A., 2024. Multimodal fusion of optimized GRU–LSTM with self-attention layer for hydrological time series forecasting. Water Resources Management, 38(15), pp.6045-6062.

[29] Z. Fu, H. Yang, A.M.C. So, W. Lam, L. Bing, N. Collier, On the effectiveness of parameter-efficient fine-tuning, Proc. AAAI Conf. Artif. Intell. 37 (2023) 12799–12807.

[30] P. Dhal, C. Azad, A fine-tuning deep learning with multi-objective-based feature selection approach for the classification of text, Neural Comput. Appl. 36 (2024) 3525–3553.