AIMS Mathematics, 11(5): 12548–12579.
DOI: 10.3934/math.2026516
Received: 22 February 2026
Revised: 20 April 2026
Accepted: 22 April 2026
Published: 07 May 2026

http://www.aimspress.com/journal/Math

Research article

A hybrid VMD-PermEn-DBSCAN-ICEEMDAN deep learning framework for ultra-short-term wind speed prediction

Musaed Alrashidi*

Department of Electrical Engineering, College of Engineering, Qassim University, Buraydah 52571, Saudi Arabia

* **Correspondence:** Email: malrashidi@qu.edu.sa.

Abstract: Accurate ultra-short-term wind speed forecasting is essential for the reliable integration of wind energy into power grids; nevertheless, it is challenging due to the non-linearity and non-stationarity of wind signals. Therefore, this research introduces a novel multi-phase hybrid framework—VMD-PermEn-DBSCAN-ICEEMDAN—aimed at improving prediction accuracy through the systematic refinement of complex signal components. The process initiates with variational mode decomposition (VMD) to decompose raw wind speed data into intrinsic mode functions (IMFs). Permutation entropy (PermEn) is employed for feature extraction to address the complexity of these components, followed by density-based spatial clustering of applications with noise (DBSCAN) clustering to categorize IMFs exhibiting analogous dynamic patterns. A secondary decomposition phase employing enhanced complete ensemble empirical mode decomposition with adaptive noise (ICEEMDAN) is aimed primarily at high-frequency clusters to reveal hidden fluctuations. These refined features serve as inputs for advanced deep learning models, such as long short-term memory (LSTM) networks, gated recurrent units (GRU), and their hybrid configurations. The framework was assessed utilizing wind speed data from Riyadh, Saudi Arabia, gathered at 5-minute intervals. Experimental findings indicated that the LSTM-GRU hybrid model consistently surpasses independent architectures and conventional machine learning methods, including artificial neural networks, support vector machines, and decision trees. The proposed framework attained a remarkable mean squared error of 0.00081 m^2/s^2 and a coefficient of determination of 0.99954 for 5-minute forecasts. In addition, the study examined the effects of input lag lengths and forecasting resolutions of up to one hour, validating the model's durability and exceptional performance in ultra-short-term scenarios. The results underscore the effectiveness of integrating adaptive signal decomposition, intelligent clustering, and deep learning for accurate wind speed prediction, offering a dependable resource for energy management and grid stability.

Keywords: time series forecast; wind speed; deep learning; signal decomposition methods; clustering; energy systems
Mathematics Subject Classification: 68T07

1. Introduction

Wind energy has emerged as a vital component of the global renewable energy sector, offering a sustainable and environmentally beneficial alternative for conventional fossil fuels. The rapid growth and increased integration into national power systems emphasize the critical need for accurate and reliable wind speed forecasts. Precise wind speed forecasts are essential for improving the operational efficiency of wind farms, facilitating effective grid management, and ensuring the stability and security of power supply [1]. Furthermore, accurate forecasts enable energy traders to make informed judgments, reduce financial risks associated with market volatility, and enhance the overall economic viability of wind-producing projects. The intermittent and stochastic nature of wind presents significant challenges to forecasting, as wind speed time series are inherently non-linear, non-stationary, and influenced by several complex atmospheric phenomena [2]. This inherent variability necessitates the creation of advanced forecasting techniques capable of capturing these complex dynamics to improve prediction reliability and accuracy.

Conventional statistical and physical forecasting techniques, although fundamental, often inadequately address the complexities inherent in wind speed data [3]. Statistical techniques, such as autoregressive (AR) [4], autoregressive moving average (ARMA) [5], and autoregressive integrated moving average (ARIMA), rely on the assumptions of linearity and stationarity, which are occasionally violated by actual wind speed patterns [6]. On the other hand, numerical weather prediction (NWP) employs physical models that yield high accuracy for longer predictions; however, they are computationally intensive and may face uncertainties related to initial conditions and boundary layer parameterizations [7]. The limitations of conventional methods have necessitated extensive exploration of hybrid models that amalgamate the benefits of many methodologies to address certain shortcomings. The advent of advanced signal decomposition methods and deep learning algorithms has facilitated the development of more robust and flexible forecasting systems, capable of handling the non-linear and non-stationary characteristics of wind speed time series with greater effectiveness [8]. The enhancement of forecasting technologies is crucial for the continuous development and efficient integration of wind energy into the global energy portfolio.

Regarding the forecasting horizons, wind speed forecasting is classified according to their temporal scope, each performing certain operational and strategic functions within the energy sector. Ultra-short-term forecasts, generally covering a duration of minutes to an hour, are essential for real-time grid management, frequency regulation, and immediate power dispatch decisions, directly influencing the stability and reliability of the electrical network [9,10]. Short-term predictions, ranging from one hour to many days, are crucial for unit commitment, economic dispatch, and energy trading, facilitating efficient scheduling of power generation and market engagement [11]. Medium-term predictions, spanning one week to one month, guide maintenance scheduling, resource allocation, and risk management techniques for wind farm administrators. Ultimately, long-term forecasts, spanning months to years, are essential for strategic planning, investment decisions, infrastructure development, and evaluating the long-term sustainability of wind energy facilities [12]. The selection of forecasting horizon profoundly affects the approaches utilized, as shorter horizons typically depend on detailed

data and advanced signal processing, whilst longer horizons integrate meteorological models and climatic trends [13]. Each horizon offers distinct difficulties and opportunities for enhancing wind power integration and securing a robust energy supply.

1.1. Related work

To address the issues presented by the non-stationary and non-linear characteristics of wind speed data, numerous signal decomposition approaches have been progressively utilized in forecasting models. These methods employ signal decomposition algorithms to partition non-stationary wind speed data into multiple relatively stable subcomponents. This approach mitigates irregularities in wind sequences and prevents interference between modes. Wavelet decomposition (WD) [14,15], empirical mode decomposition (EMD) [16], ensemble empirical mode decomposition (EEMD) [17], complete ensemble empirical mode decomposition (CEEMD) [18], and complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) are prevalent methodologies for signal decomposition in wind speed forecasting [19]. Recently, variational mode decomposition (VMD) has developed as a productive and adaptable method for decomposing complicated signals into a small set of quasi-orthogonal intrinsic mode functions (IMFs) [20]. In contrast to conventional techniques like EMD and EEMD, VMD is non-recursive and is supported by a robust theoretical framework based on variational optimization, effectively addressing challenges such as mode mixing and spurious modes [21]. This attribute renders VMD especially adept at evaluating highly variable time series such as wind speed. Various research has proven the efficacy of VMD in improving the precision of wind speed forecasts through the preprocessing of raw data. Research also indicates that decomposing wind speed signals using VMD prior to inputting them into forecasting models markedly enhances performance compared to utilizing raw data directly [22]. In addition to these methods, enhanced complete ensemble empirical mode decomposition with adaptive noise (ICEEMDAN) has developed as an effective signal processing methodology. ICEEMDAN addresses the shortcomings of previous EMD variations by integrating white noise at each decomposition phase and calculating the local means of the relevant IMFs, resulting in a more comprehensive and physically significant decomposition. This method is especially efficient in generating IMFs that are more refined and physically interpreted, mitigating problems such as mode mixing and spurious components that sometimes affect decomposition strategies [23]. This technique makes it exceptionally beneficial for the analysis of complicated and noisy time series data, such as wind speed, hence enhancing the quality of inputs for subsequent forecasting models.

Several studies have used these advanced signal decomposition methods. For example, Li et al. [24] developed a hybrid model for short-term power load forecasting that integrates VMD with an extreme learning machine (ELM) to tackle the non-linear and non-stationary characteristics of power load data. Their process includes decomposing the original series into several modes by VMD, forecasting each mode with an ELM, and improving critical parameters based on forecasting precision. The ELM-VMD model exhibited high performance, attaining a mean absolute percentage error (MAPE) of 0.0107%, a root mean square error (RMSE) of 2.94 MW, and a coefficient of determination (R^2) of 0.998. In addition, Wang et al. [25] introduced a wind speed prediction model that integrates enhanced VMD for noise reduction, long short-term memory (LSTM) neural networks for deterministic forecasting, and a corrective mechanism utilizing improved particle swarm optimization (PSOR) based on wind speed ramp events. Their methodology incorporates an interval prediction strategy utilizing the Lorenz disturbance sequence and B-spline interpolation to assess variations in wind speed. Experimental findings from two real wind datasets indicate that the suggested VMD-LSTM-PSOR model markedly

diminishes forecasting errors (e.g., MAPE as low as 4.0% on test sets) and attains superior interval coverage rates relative to benchmark models. On the other hand, the study in [26] developed a hybrid model for wind speed forecasting that combines ICEEMDAN, multiscale fuzzy entropy (MFE), LSTM, and the informer model. The process entails decomposing wind speed data into IMFs by ICEEMDAN, aggregating components through MFE, and subsequently using LSTM and an informer to forecast each subsequence according to its complexity. Comparative assessments on two real-world datasets demonstrated that the ICEEMDAN-MFE-LSTM-INFORMER model attains the maximum accuracy among eight evaluated models, exhibiting reduced RMSE, mean absolute error (MAE), and MAPE values. Therefore, the implementation of VMD in this study is a crucial preliminary phase in developing dependable and accurate wind speed prediction systems. However, VMD requires careful setup, and the resultant components may vary in complexity, necessitating further refinement. Hence, ICEEMDAN is employed in this work to further refine the complex components of VMD to reveal the hidden information.

Furthermore, several studies have incorporated entropy-based methods for assessing signal complexity, which improves the interpretability and efficiency of decomposed signals. Permutation entropy (PermEn) is an especially useful metric owing to its resilience to noise and computational efficiency. PermEn offers a quantitative evaluation of randomness by analyzing the ordinal patterns within each IMF. This information is essential for distinguishing components that signify significant physical trends from those that are predominantly stochastic noise, thereby informing the following processing steps and alleviating the computational load on forecasting models. For example, the authors in [27] investigated the inherent predictability of wind speed time series to enhance wind energy resource evaluations and grid stability. They used entropy-based methods, namely PermEn and sample entropy (SampEn), to measure signal complexity under various settings, including terrain intricacy and seasonal fluctuations. The findings demonstrate that the proposed framework efficiently differentiates between deterministic and stochastic dynamics, resulting in a significant enhancement in prediction accuracy—up to 64.33% in MAPE when combined with hybrid deep learning models. On the other hand, Ruiz-Aguilar et al. [28] proposed a hybrid methodology for wind speed forecasting that combines EMD, permutation entropy (PE) for component selection, and artificial neural networks (ANNs) within an ensemble framework. Their methodology decomposes wind speed time series into IMFs using EMD, uses PE to recognize and aggregate less-complex components, and then forecasts each chosen component with ANNs. The EMD-PE-ANN ensemble demonstrated enhanced accuracy compared to individual ANN models, with an R^2 of 0.981 for 1-hour predictions and 0.807 for 24-hour predictions, indicating significant advancements, particularly for lengthy forecasting periods. Nevertheless, PermEn may fail to differentiate between structurally analogous dynamical patterns at varying scales, potentially resulting in redundant or noisy feature representations and diminished interpretability in complex wind speed signals.

Clustering algorithms provide a systematic method to handle the large number of components produced during signal decomposition [29]. Density-based spatial clustering of applications with noise (DBSCAN) is particularly appropriate for this objective, as it does not necessitate a predetermined number of clusters and can efficiently detect noise points. DBSCAN streamlines the signal structure by clustering IMFs with analogous dynamic patterns utilizing statistical and entropy-based characteristics. This classification facilitates a more focused implementation of subsequent decomposition methods, guaranteeing that only the most complex clusters undergo further refinement while maintaining a consistency of more uniform trends. Zhang et al. [30] presented a short-term wind power forecasting model that integrates outlier detection and correction through the DBSCAN method with a genetic algorithm-optimized back propagation (GA-BP) neural network. Their methodology

employs DBSCAN for outlier detection and linear regression for correction in wind speed and power data, identifies analogous days by clustering and Euclidean distance, and subsequently utilizes the refined data as input for the GA-BP neural network. Experimental results indicate that this methodology markedly enhances prediction accuracy, decreasing the normalized root mean square error (NRSME) to 6.46% and the normalized mean absolute error (NMAE) to 5.24%, exceeding models lacking data preprocessing. Another study in [31] introduced a short-term wind power forecasting technique that integrates DBSCAN clustering, principal component analysis (PCA), and support vector machine regression (SVR) to tackle the dynamic variability of training data and enhance prediction precision. Their methodology employs PCA for dimensionality reduction, DBSCAN for clustering similar days based on meteorological and historical power data, and SVR for regression, using the most important clusters as the training dataset. Experimental results indicate that the DBSCAN-SVR approach markedly surpasses conventional SVR, diminishing RMSE by 43.1% and MAE by 54.5%.

1.2. Motivation and contributions of the study

Based on the above discussion, the amalgamation of these advanced signal processing and clustering methodologies establishes an effective preprocessing pipeline, converting raw, complex wind speed data into a collection of refined, more predictable components. Therefore, this paper proposes a novel hybrid framework—VMD-PermEn-DBSCAN-ICEEMDAN—integrated with deep learning models for ultra-short-term wind speed forecasting. This thorough preprocessing strategy integrates VMD for the initial decomposition of wind speed data, PermEn for feature extraction, DBSCAN for intelligent clustering, and ICEEMDAN for further refinement of high-frequency components, with the objective of substantially improving the quality of input data for predictive models. The primary aim of this multi-stage framework is to enhance the precision and dependability of wind speed forecasting, which is essential for the effective operation and integration of wind energy systems. The primary contributions of this study in relation to existing literature are as follows:

- Development of a multi-stage hybrid framework that integrates VMD, PermEn, DBSCAN clustering, and ICEEMDAN for advanced signal preprocessing.
- Comprehensive evaluation of deep learning architectures, specifically identifying the LSTM-GRU hybrid configuration as the most effective model for capturing refined temporal features.
- Extensive comparative analysis against standalone deep learning models, conventional machine learning techniques (ANN, SVM, DT), and simplified hybrid models, demonstrating the superior accuracy of the proposed framework.
- Investigation of the impact of input lag and forecasting resolutions, providing insights into the optimal configurations for ultra-short-term wind speed prediction.

The following sections of this paper are organized as follows: Section 2 describes the proposed multi-stage hybrid framework, and data acquisition and preparation. Section 3 describes the results and discussion, including a comparative analysis of deep and machine learning performance, an evaluation of input data sizes, and the proposed method's performance across different forecasting resolutions. Finally, Section 4 concludes the study, examining the limitations of this research and offering prospective avenues for future research.

2. Materials and methods

This paper presents a multi-stage hybrid framework for precise wind speed forecasting, incorporating advanced signal decomposition, feature engineering, clustering, and deep learning methodologies. In contrast to prior methods that often employ decomposition techniques in isolation or utilize fundamental clustering procedures, this research develops a unique adaptive clustering-refinement mechanism. The method utilizes PermEn to derive significant features from VMD components, and then employs DBSCAN clustering to categorize IMFs according to their dynamic patterns. This enables the accurate and focused implementation of ICEEMDAN on high-frequency, intricate clusters, revealing subtle variations that are frequently disregarded. This adaptive and hierarchical preprocessing pipeline, combined with the enhanced predictive capabilities of a hybrid deep learning model, signifies an important milestone in addressing the intrinsic non-linearity and non-stationarity of wind speed data, resulting in exceptional accuracy in ultra-short-term forecasting.

The framework is methodically organized into numerous essential processes, each aiding in the enhancement and readiness of the data for accurate forecasting. In addition, the proposed framework incorporates several preprocessing phases, each applied to a univariate signal or a limited number of decomposed components, hence maintaining computational feasibility for ultra-short-term forecasting applications. Figure 1 shows the overall framework of the proposed wind speed forecasting model. The framework is implemented using the following steps:

- *Step 1:* Data Preprocessing. The framework begins by importing and preparing the raw wind speed data to guarantee its appropriateness for further analysis. This involves dealing with missing values and normalizing the data. Missing values are tackled by linear interpolation to preserve the continuity of wind speed data.
- *Step 2:* Initial Signal Decomposition using VMD. The wind speed data are decomposed into different IMFs using the VMD method. The procedure guarantees a more resilient and physically significant separation of modes, which reflect different oscillatory components of the original signal.
- *Step 3:* Feature Extraction with PermEn. For each IMF obtained in *Step 2*, the PermEn is calculated based on the ordinal patterns of its values, yielding a numerical value that indicates the regularity or randomness of the component.
- *Step 4:* IMF Clustering using DBSCAN. DBSCAN is utilized to cluster the IMFs according to their acquired features, which encompass normalized PermEn, mean, standard deviation, kurtosis, and skewness. The clustering step seeks to unify components exhibiting analogous dynamic patterns, thereby diminishing the complexity of the decomposed wind speed signals.
- *Step 5:* Secondary Decomposition using ICEEMDAN. ICEEMDAN is used to further assist in refining the complex IMF component and capturing the hidden fluctuation in the wind speed signals.
- *Step 6:* Forecasting Models' Configurations. All refined IMFs are then exposed to the forecasting models. In this study, deep learning models are employed for their ability to identify complex temporal correlations and non-linear interactions within the preprocessed data. The models are LSTM networks, gated recurrent units (GRU), and their hybrid configurations (LSTM-GRU and GRU-LSTM).
- *Step 7:* Developing Training Models. The designed deep learning models are trained using the training data set. Each of the forecasting models is exposed to the same training and testing data sets.
- *Step 8:* Performance Evaluation: The models developed in *Step 7* are evaluated using the testing data set. The evaluation metrics used include the mean squared error (MSE), coefficient of determination (R^2), root mean square error (RMSE), normalized root mean square error (nRMSE),

mean absolute error (MAE), and normalized mean absolute error (nMAE).

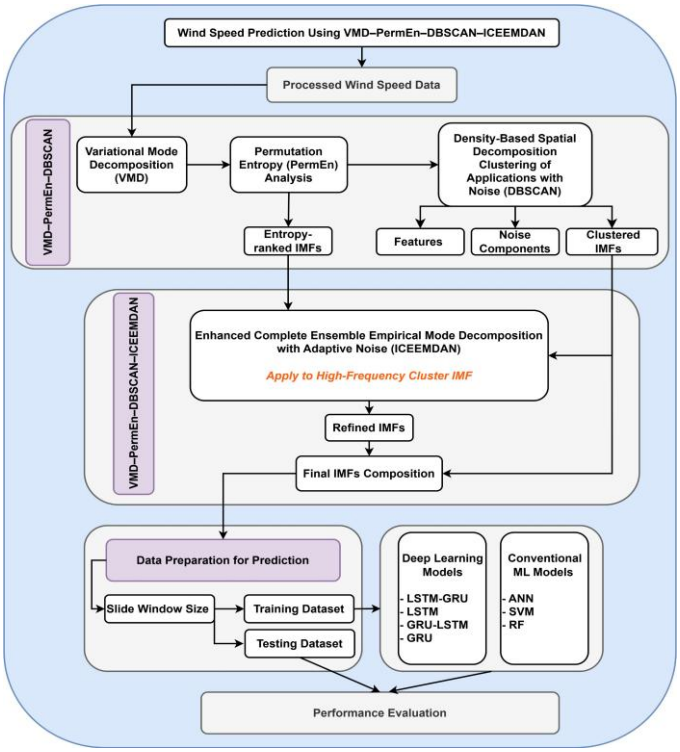


Figure 1. The hybrid VMD-PermEn-DBSCAN-ICEEMDAN framework for wind speed prediction.

The subsequent subsections explain each element of the proposed methodology, encompassing the foundational ideas and relevant mathematical formulations.

2.1. Data acquisition and preparation

2.1.1. Data acquisition

Solcast is a company that provides global solar irradiance statistics. Researchers can acquire significant data, and the public can access this information freely through https://solcast.com/. Solcast can provide multiple meteorological variables across several time intervals (5, 30, and 60 minutes), including global horizontal irradiance (GHI), diffuse horizontal irradiance (DIF), direct normal irradiance (DNI), air temperature, solar zenith angle, solar azimuth angle, cloud cover, atmospheric pressure, wind speed, and wind direction. In this study, the wind speed data are measured in Riyadh, Saudi Arabia, at the following coordinates: latitude: 24.90689°N and longitude: 46.39721°E; see Figure 2. Wind speed data are gathered at a height of 10 meters and recorded at 5-minute intervals from January 1, 2022, to December 31, 2022. The mean wind speed for 2022 is 3.626 m/s, while the peak recorded by the anemometer occurred on April 25th, 2022, at 10:05 A.M., at 11.2 m/s.

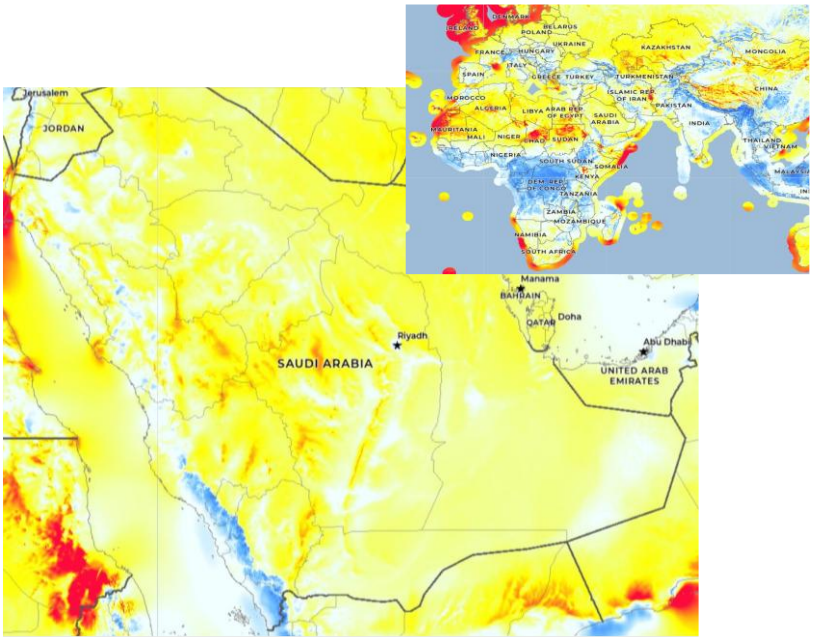


Figure 2. Saudi wind speed map. The map is collected from the Global Wind Atlas (https://globalwindatlas.info).

2.1.2. Data preparation

The first stage involves importing the raw wind speed data and applying necessary preprocessing to guarantee data quality and uniformity. The values that are missing in the wind speed data are addressed by linear interpolation. After that, the wind speed data is standardized using Z-score normalization. This procedure normalizes the data to achieve a mean of zero and a standard deviation of one, which is essential for numerous signal processing and forecasting algorithms, as it prevents characteristics with greater numerical ranges from overshadowing the learning process. The formula for normalization is expressed as follows:

$$X_{normalized} = \frac{X-\mu}{\sigma}, \tag{1}$$

where X denotes the original wind speed data, and μ and σ signify the mean and standard deviation of the wind speed data, respectively. The normalization statistics (μ and σ) are calculated from the training data and subsequently applied to the validation and testing subsets to guarantee a leakage-free scaling process. This step guarantees that the data is formatted appropriately for the next decomposition and analysis phases, enhancing numerical stability and optimizing model performance.

2.2. Variational mode decomposition (VMD)

VMD serves as the principal method for signal decomposition to tackle the non-stationarity and non-linearity present in wind speed data. VMD was introduced in 2014 by Dragomiretskiy and Zosso [20], and it adaptively decomposes a complex signal into a finite number of quasi-orthogonal IMFs, each defined by a specific center frequency and bandwidth. This method is resilient to noise

and mode mixing, providing a superior alternative to conventional EMD techniques [32]. The fundamental attribute of IMFs in VMD is that each exhibits a cosine function waveform, characterized by slowly varying and positive envelopes, alongside an instantaneous frequency that changes gradually in a nondecreasing manner. The process of signal decomposition with VMD encompasses Wiener filtering, Hilbert transformation, frequency mixing, and heterodyne demodulation. The basic method involves identifying a distinct set of IMFs u_k and their corresponding central frequencies ω_k that minimize the constrained variational problem shown in Eqs (1) and (2) [20,33].

$$min_{\{u_k\},\{\omega_k\}} \left\{\sum_{k=1}^{K} ||\partial_t[\left(\delta(t)+\frac{j}{\pi t}\right) * u_k(t)]\, e^{-j\omega_k t}||_2^2\right\}, \tag{2}$$

$$subject\ to\ \sum_{k=1}^{K} u_k = f(t), \tag{3}$$

where u_k denotes the k-th IMF, ω_k represents its center frequency, K signifies the total number of IMFs, $\delta(t)$ indicates the Dirac delta function, j symbolizes the imaginary unit, $*$ refers to the convolution operator, and $f(t)$ represents the original signal. The augmented Lagrangian approach is employed to address this constrained optimization problem, incorporating a quadratic penalty component and Lagrangian multipliers [34]. The augmented Lagrangian $\mathcal{L}$ is defined as follows:

$$\begin{aligned}\mathcal{L}(\{u_k\},\{\omega_k\},\lambda) = \alpha\sum_{k=1}^{K} \left\|\partial_t\left[\left(\delta(t)+\frac{j}{\pi t}\right) * u_k(t)\right] e^{-j\omega_k t}\right\|_2^2 + \|f(t) - \\ \sum_{k=1}^{K} u_k(t)\|_2^2 + \left\langle\lambda(t), f(t) - \sum_{k=1}^{K} u_k(t)\right\rangle\end{aligned} \tag{4}$$

where α represents the balancing parameter of the data fidelity constraint and λ denotes the Lagrangian multiplier. The optimization is conducted repeatedly by updating u_k, ω_k, and λ in the frequency domain by the alternating direction method of multipliers (ADMM). The quantity of parameter K is a very crucial step. This work configures the VMD to decompose the original wind speed signal into 12 IMFs, as depicted in Figure 3. Each IMF denotes a distinct frequency band, encompassing high-frequency variations to low-frequency trends, facilitating a comprehensive examination of the wind speed signal across many scales. The choice of K = 12 yielded a steady breakdown of the wind-speed signal into comprehensible frequency bands, facilitating subsequent entropy assessment and cluster differentiation without apparent under-decomposition or unnecessary mode division.

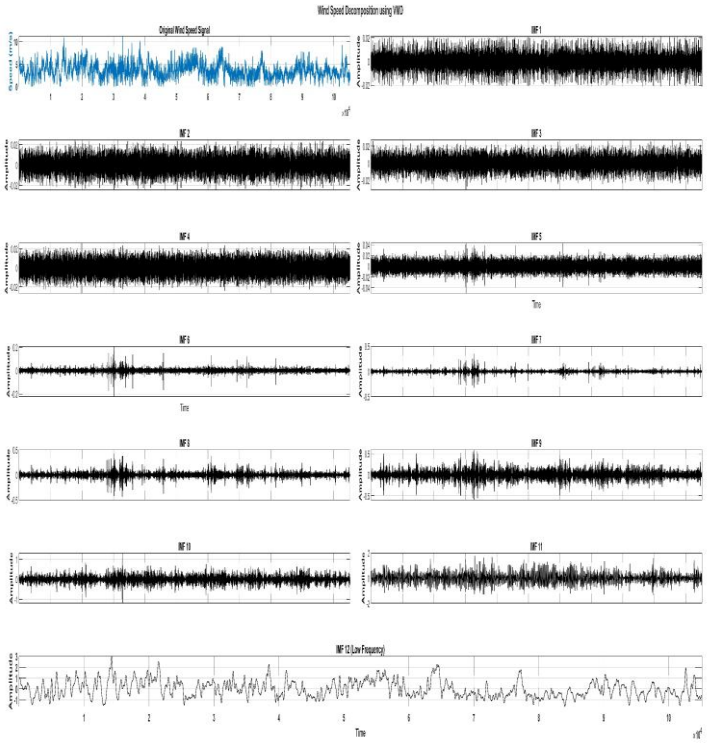


Figure 3. Decomposition results of the wind speed data into IMFs using VMD.

2.3. Permutation entropy (PermEn)

After the decomposition of the wind speed signal into IMFs by VMD, PermEn is applied for each IMF to assess its complexity and randomness. PermEn was proposed in 2002 by Christoph Bandt and Georg Pompe [35]. PermEn is a resilient and computationally efficient metric that reflects the ordinal patterns in time series data, rendering it less susceptible to amplitude fluctuations and noise. In this study, PermEn is utilized due to its efficient and robust capture of ordinal complexity, rendering it especially adept at differentiating predictable IMF structures from highly irregular ones in noisy wind-speed signals. The method utilizes ordinal patterns, permutations that denote the relative order of data points, and computes entropy based on their probability distribution [36]. The computation of PermEn entails multiple stages as follows [37,38]:

1) Phase Space Reconstruction: For a specified time series data $X = \{x_1, x_2, \dots, x_N\}$, a collection of embedding vectors Y_j is formulated as follows: $Y_j = [x_j, x_{j+\tau}, \dots, x_{j+(m-1)\tau}]$, where m denotes the embedding dimension and τ represents the time delay. In this study, m and τ are set to be 3 and 1, respectively.

2) Ranking and Permutation: Each embedding vector Y_j is subsequently transferred to a permutation $\pi_j = (r_0, r_1, \dots, r_{m-1})$ of $(0, 1, \dots, m-1)$ by ranking its elements in increasing sequence. For instance, $x_{j+r_0\,\tau} < x_{j+r_1\,\tau} \leq \cdots \leq x_{j+r_{m-1}\,\tau}$. If two values are equivalent, their original indices are employed to resolve ties.

3) Probability Distribution: The probability $P(\pi_j)$ of each distinct permutation π_j in the time series is determined by listing its occurrences and dividing by the total number of embedding vectors, $P(\pi_j) = \frac{\text{Number of occurrences of } \pi_j}{(N-(m-1)\tau)}$.

4) Calculation of Shannon Entropy: The Shannon entropy $H_p(m)$ is calculated as follows:

$H_p(m) = -\sum_\pi P(\pi)\, ln\, P(\pi)$, where the summation encompasses all $(m!)$ possible permutations. The maximal entropy is $ln(m!)$ when all permutations are equally probable.

5) Normalization: The normalized permutation entropy $PE_{normalized}$ is derived by dividing $H_p(m)$ by its maximum attainable value: $PE_{normalized} = \frac{H_p(m)}{ln(m!)}$.

The normalized values span from 0 to 1, with values near 0 signifying highly regular or predictable data, while values near 1 imply a highly unpredictable or complex time series. In this study, both the raw and normalized PermEn for each IMF are computed, offering a quantitative assessment of their intrinsic complexity, which are subsequently utilized as a feature for clustering. The results are shown in Table 1.

Table 1. The results of permutation entropy and DBSCAN clustering of the IMFs generated by VMD.

Subsequence	Permutation Entropy	DBSCAN Clustering
IMF 1	1.7600	1
IMF 2	1.7391	1
IMF 3	1.7210	1
IMF 4	1.7189	1
IMF 5	1.5359	1
IMF 6	1.2223	2
IMF 7	1.0192	2
IMF 8	0.9196	3
IMF 9	0.8557	3
IMF 10	0.8075	3
IMF 11	0.7827	-1
IMF 12	0.7663	-1

2.4. DBSCAN clustering

DBSCAN was first proposed by M. Ester et al. [39]. In this study, it is utilized to cluster the IMFs according to their extracted characteristics, which encompass normalized PermEn, mean, standard deviation, kurtosis, and skewness. This clustering phase seeks to amalgamate components exhibiting analogous dynamic patterns, thereby diminishing complexity and enhancing the interpretability of the deconstructed signals. DBSCAN is especially appropriate for this task, as it does not necessitate a predetermined number of clusters and is capable of detecting noise points. The principal parameters for DBSCAN are [30]:

- Epsilon (ϵ): The greatest distance between two samples for one to be regarded as within the vicinity of the other. In this study, ϵ is equal to 1.5.
- Minimum Points $(MinPts)$: The requisite number of samples (or cumulative weight) inside a neighborhood for a point to qualify as a core point. $MinPts$ is set to be 2 in this study.

According to these factors, DBSCAN categorizes points into three classifications:

- Core Point: A point that possesses a minimum of $MinPts$ points (including itself) within its neighborhood.
- Border Point: A point that possesses fewer than $MinPts$ points inside its ϵ-neighborhood yet

resides within the ϵ-neighborhood of a core point.

- Noise Point: A point that is neither classified as a core point nor a border point.

The DBSCAN parameters are implemented as they provided stable density-based separation of the IMFs in the normalized feature space and generated clusters that aligned with the complexity ranking indicated by PermEn. In this study, the extracted features from each IMF form a feature vector, which is normalized using the Z-score before using DBSCAN to ensure that all features contribute equally to the distance calculation. DBSCAN is implemented in the original five-dimensional normalized feature space, while PCA is utilized solely for the presentation of the resultant clustering structure. In addition, Algorithm 1 describes the DBSCAN algorithm, while Table 1 lists the PermEn values and DBSCAN clustering for each IMF generated by VMD. It can be noticed that the ordered components can be clustered into three groups, namely, IMF_1, IMF_2, and IMF_3, while IMF_{-1} is set for the IMFs that are clustered as noise components.

Algorithm 1. DBSCAN $(D, \varepsilon, MinPts)$

———— Input ————

P: set of data points

ε: neighborhood radius

$MinPts$: minimum number of points required to form a dense region (core point)

———— Main Algorithm ————

Initialize cluster index $C = 0$
Initialize the set of clusters as empty: $Clusters = \emptyset$
For all $p \in P$ that has not yet been visited do
    Mark p as visited
    Find all points within ε-distance of p:
    $NeighborSet = regionQuery(p, \varepsilon)$
    If the number of points in $NeighborSet$ is less than $MinPts$ then
        Label p as $noise$
    Else
    Increment cluster counter: $C = C + 1$
    Create a new cluster by expanding from p
    $Clusters = Clusters \cup expandCluster(p, NeighborSet, C, \varepsilon, MinPts)$
    End if
End for
Return the set of $Clusters$

The PermEn values introduce a complexity dimension that improves separability within the feature space utilized by DBSCAN. Components exhibiting analogous entropy and statistical features constitute compact density clusters, whereas highly irregular intrinsic mode functions show as isolated noise spots. Table 1 illustrates this correlation: high PermEn values (>1.5) aggregate in Cluster 1, moderate values (1.0–1.5) constitute Cluster 2, and low values (<1.0) comprise Cluster 3. Figure 4 depicts the three-dimensional PCA-based DBSCAN clustering of 12 VMD-derived IMFs utilizing five statistical-entropy features. The IMFs are represented in the top subplot as projections onto PC1 (63%), PC2 (23%), and PC3 (11%), collectively accounting for 97% of the total variance. This verifies that the three-dimensional space retains almost all information from the original feature set. DBSCAN (ε

= 1.5, $MinPts$ = 2) creates three compact clusters (cluster 1, 2, and 3) and classifies two IMFs (IMF 11 and IMF 12) as noise (cluster−1). The noise IMFs are spatially segregated from the dense clusters in the PCA space. The bottom-left subplot displays these noise IMFs, distinguished by unpredictable, high-frequency, and stochastic variations with significant amplitude variability. Hence, the clustering suggests no physical significance for prediction. Conversely, the bottom-right subplot displays the clustered IMFs, which demonstrate smoother and more organized oscillatory patterns, reflecting significant wind-speed dynamics across many temporal scales.

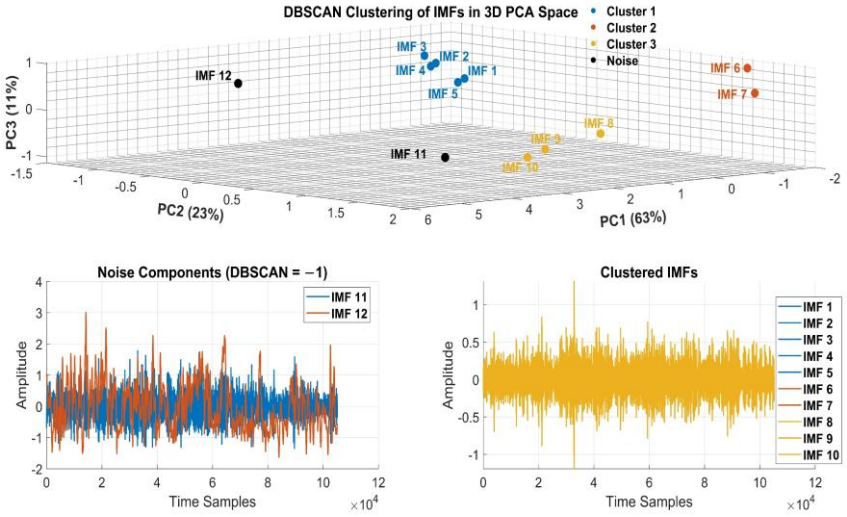


Figure 4. The 3D PCA-based DBSCAN clustering of 12 VMD-derived IMFs.

2.5. Enhanced complete ensemble empirical mode decomposition with adaptive noise (ICEEMDAN)

The ICEEMDAN signal processing approach was introduced by M. Colominas in 2014 [40]. It is utilized in this study to enhance the breakdown of complex components. ICEEMDAN is applied exclusively on the high-frequency/complex cluster (IMF 1 in Cluster 1), as this cluster exhibits the most irregular oscillatory activity, while smoother clusters possess an orderly predictive structure and so do not require additional refinement. ICEEMDAN is a sophisticated, noise-assisted technique for signal decomposition designed to address the limitations of conventional EMD and EEMD. This is accomplished by incorporating specific white noise into the signal at each decomposition stage and thereafter calculating local means of the relevant IMFs, yielding a more comprehensive and physically significant decomposition. The ICEEMDAN algorithm can be outlined as follows [41,42]:

1) In each iteration i, a noisy variant of the signal x is produced by incorporating a particular realization of white Gaussian noise $\omega^{(i)}$, scaled by a noise standard deviation coefficient β_0:

$$x_1^{(i)} = x + \beta_0 \cdot E_1(\omega^{(i)}), \tag{5}$$

where $x_1^{(i)}$ is the first mixed signal and $E_1(\cdot)$ signifies the operator utilized for computing the initial IMF component.

2) Calculate and decompose the initial mixed signal $x_1^{(i)}$ using EMD across K realizations and compute the first IMF. This step is as follows:

$$IMF_1 = x - r_1, \tag{6}$$

$$r_1 = \langle M(x_1^{(i)}) \rangle, \tag{7}$$

where r_1 denotes the first residue, $\langle \cdot \rangle$ is the average operator, and $M(\cdot)$ represents the local mean.
3) The second mixed signal $x_2^{(i)}$ can be calculated as follows:

$$x_2^{(i)} = r_1 + \beta_1 E_2(\omega^{(i)}). \tag{8}$$

Decompose the second mixed signal $x_2^{(i)}$ using EMD across *K* realizations and compute the IMF_2 accordingly:

$$IMF_2 = r_1 - r_2, \tag{9}$$

$$r_2 = \langle M(x_2^{(i)}) \rangle. \tag{10}$$

4) Execute the previous steps for the *k*-th decomposition to derive the *k*-th IMF, continuing until the stopping condition is met:

$$IMF_k = r_{k-1} - \langle M\left[r_{k-1} + \beta_{k-1} E_k(w^i)\right] \rangle. \tag{11}$$

2.6. Data partitioning for training, testing, and evaluation

A sliding window strategy is employed to prepare the data for time series prediction by constructing consecutive inputs and their associated target outputs. This technique is essential for training the deep learning algorithms and other machine learning models that necessitate a sequence of historical observations to forecast future values. This study employs multiple lags of wind speed data: 5, 10, 15, 20, and 25 previous wind speed readings. For instance, 20 means that the preceding 20 time steps of the final IMF components are utilized to forecast the next wind speed value. The selected lag range is intended to encompass practical short-memory intervals for ultra-short-term forecasting, and their comparative impact is then assessed in Section 3.3. Following the construction of these sequences, the data is partitioned into training, validation, and testing sets to assess the model's generalization capacity. A ratio of 0.7:0.1:0.2 is utilized, signifying that 70% of the data is allocated for training, with 10% and 20% assigned for validation and testing, respectively. The data partition is executed chronologically, with each sliding window utilizing solely historical observations prior to the forecasting target, thus avoiding temporal leaking. In addition, for each forecasting horizon, the model uses a direct multi-step approach, producing predictions for each horizon individually instead of recursively using prior predictions. Algorithm 2 explains the training and testing phase used in this study.

Algorithm 2. Training and testing phase

———— Training Phase ————

Forecasting resolution settings: *(FR = 5 min, 15 min, 30 min, 45 min, and 1 hour)*
Lag observations of refined IMF settings: *(Lag = 5 min, 10 min, 15 min, 20 min, and 25 min)*
Data loading: *(M1)*
Load the output day: *(D = 288 × 1)*
Apply *Slide Window technique* to divide *(M1)* and *(D)* using *(FR)* and *(Lag)*
Mark *(Train)* as training dataset
Mark *(Test)* as testing dataset
Mark *(Valid)* as validation dataset
Split the target (T_{WS}) into (T_{Train}), (T_{Test}), and (T_{Valid}) for training, testing, and validation
for each Algorithm *(DL),* **do**
    Train *(DL)* using *(Train*) as input and (T_{Train}) as output
    Validate *(DL)* using *(Valid)* as input and (T_{Valid}) as output
    Save the trained deep learning model as (M_{DL})
end for

———— Testing Phase ————

Load *(DL)*, *(Test)*, (T_{Test})
for each trained model (M_{DL}), **do**
    Test (M_{DL}) using *(Test)* as input
    Save the predicted output (T_{DL})
    Compare (T_{DL}) and (T_{Test}) and save the results
end for

2.7. Deep learning models

This study integrates multiple advanced deep learning architectures for wind speed prediction, utilizing their capacity to extract intricate temporal dependencies and non-linear interactions within the preprocessed data. The models include LSTM, GRU, and their hybrid configurations (LSTM-GRU and GRU-LSTM). RNNs are especially adept at time series forecasting because of their internal memory processes, which enable them to learn and retain patterns across extended periods.

- LSTM: LSTMs are a distinct type of RNN adept at acquiring long-term dependencies. They accomplish this using an intricate architecture consisting of a cell state and three gates: the input gate, the forget gate, and the output gate; see Figure 5(a). These gates control the influx and efflux of information within the cell, enabling the network to selectively retain or discard information over time [43].

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f), \tag{12}$$

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i), \tag{13}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot tanh(W_c[h_{t-1}, x_t] + b_i), \tag{14}$$

$$ht = O_t \tanh \odot (c_t). \tag{15}$$

- GRU: GRUs are a more straightforward variation of LSTMs, engineered to be computationally more efficient while effectively capturing long-term dependencies [44]. They amalgamate the forget

and input gates into a singular entity; see Figure 5(b).

- Hybrid Architectures (LSTM-GRU, GRU-LSTM): These models include layers of LSTM and GRU units to exploit the advantages of both designs, facilitating enhanced feature extraction and temporal pattern recognition. The precise configurations of these hybrid models (e.g., layer count, units per layer, activation functions) are established during the model definition process.

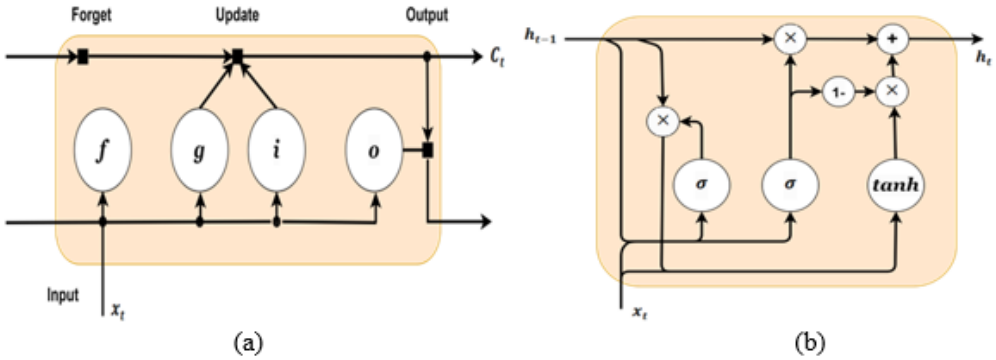


Figure 5. (a) The LSTM cell structure; (b) the GRU cell structure.

2.8. Conventional machine learning models

The proposed model integrates various conventional machine learning methods alongside deep learning architectures for comparative analysis. These models function as benchmarks to evaluate the performance enhancements attained using the hybrid decomposition and deep learning methodology. This paper examines standard machine learning techniques, including ANNs, SVMs, and DTs.

2.8.1. Support vector machines

SVMs are a widely utilized supervised learning method for classification, regression, and anomaly detection problems. The primary goal is to determine the optimal hyperplane that can classify the data into distinct categories. The method operates by dividing the training dataset into distinct groups, thereby enhancing the separation among them. Unique data is evaluated and classified into the nearest category based on precise estimations. For linearly separable data, the approach employs a hyperplane; conversely, for non-linear data, it utilizes various kernel functions based on the data type [45].

2.8.2. Artificial neural networks

ANNs are computer frameworks intended to replicate the structure and operation of biological neural networks seen in the human brain. They consist of interconnected layers of nodes or neurons that process data by passing signals through weighted connections [46]. ANNs can comprehend complex structures and relationships within data, making them suitable for diverse tasks, including classification, regression, pattern recognition, and prediction.

An ANN comprises an input layer for data collecting, one or more hidden layers for feature

extraction and transformation, and an output layer that generates the outcomes. Neurons in concealed layers employ activation functions—such as sigmoid, ReLU, or tanh—to introduce non-linearity, enabling the network to model complex, non-linear relationships [47]. ANNs acquire information during a training phase, wherein the network modifies the weights of its connections to reduce the discrepancy between its predictions and the actual results, frequently employing algorithms such as backpropagation in conjunction with optimization methods like gradient descent. This learning approach progressively modifies the weights across multiple epochs until the network's performance attains stability.

2.8.3. Decision trees

The DT algorithm is a predictive model that segments data according to its properties. It originates at the root node and partitions the data into smaller subsets using binary inquiries for feature values. The method continues, generating branches that result in leaf nodes, which yield the ultimate prediction or classification [48]. The tree is constructed by choosing the splits that most effectively partition the data based on a specific criterion, such as minimizing impurity or error. It functions as a flowchart that systematically directs decision-making until a conclusion is attained.

2.9. Assessment metrics

In this work, a comprehensive set of statistical indicators is utilized to systematically assess the performance of the wind forecasting models developed using various deep and machine learning algorithms. These metrics measure the precision and dependability of wind speed forecasts, offering a foundation for comparison and evaluation of model effectiveness. The following metrics are calculated [49,50]:

- Mean Squared Error (MSE): MSE quantifies the average of the squares of the errors, offering an assessment of the average magnitude of the errors. It imposes greater penalties on larger deviations.
- Coefficient of Determination (R^2): R^2 signifies the fraction of variance in the dependent variable that can be anticipated from the independent variables. An elevated R^2 value signifies a superior alignment of the model with the data.
- Root Mean Squared Error (RMSE): RMSE is a frequently employed metric for assessing the discrepancies between values forecasted by a model or estimate and the values recorded. RMSE is frequently favored over MSE due to its alignment with the units of the target variable.
- Normalized RMSE (nRMSE): nRMSE is a standardized variant of RMSE, facilitating the comparison of model performance across diverse datasets of varying sizes. It is generally represented as a percentage.
- Mean Absolute Error (MAE): MAE quantifies the average magnitude of errors in a series of forecasts, disregarding their direction. It exhibits less sensitivity to outliers in comparison to MSE or RMSE.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - f_i)^2, \tag{16}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(\tilde{y} - f_i)^2}{\sum_{i=1}^{n}(\tilde{y} - y_i)^2}, \tag{17}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - f_i)^2}, \tag{18}$$

$$nRMSE = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - f_i)^2}}{y_{i,max}}, \tag{19}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - f_i|, \tag{20}$$

$$nMAE = \frac{\frac{1}{n}\sum_{i=1}^{n}|y_i - f_i|}{y_{i,max}}, \tag{21}$$

where n is the number of wind speed readings in the testing datasets, y_i represents the actual measured values of the wind speed, $y_{i,max}$ refers to the maximum value within the testing dataset, and f_i represents the predicted values generated by the forecasting models. The average of the actual values of the wind speed, y_i, is denoted by $\tilde{y}$.

3. Results and discussion

This section highlights the results attained from the implementation of the proposed hybrid VMD-PermEn-DBSCAN-ICEEMDAN framework for wind speed prediction, accompanied by a detailed discussion of the findings. The efficacy of different deep learning and conventional machine learning algorithms, trained on the refined IMFs produced by the framework, is assessed utilizing a range of statistical criteria. The analysis is to illustrate the effectiveness of the multi-stage preprocessing method in improving prediction accuracy and to choose the best models for short-term wind speed forecasting. This study employs MATLAB R2024a as the computational program, using a simulation machine featuring an Intel Core i9-13900 CPU at 5.4 GHz, an NVIDIA GeForce RTX 5070 GPU, and 64 GB of RAM.

3.1. Performance of deep learning algorithms

In this subsection, the deep learning algorithms are evaluated. Despite the inherently obscure nature of deep learning models, the decomposition-entropy-clustering process offers an interpretable representation of the fundamental signal structure. The IMFs, complexity scores, and cluster assignments collectively provide a clear intermediate comprehension of the temporal components being modeled. In addition, the amount of training data may influence the precision and complexity of the prediction process. Therefore, 35 independent forecasting models are created to estimate future wind speed readings at 5min intervals. These models are seven forecasting algorithms—four deep learning and three machine learning—that were developed using five potential combinations of the refined IMFs obtained by the proposed method—lag of 5 min, 10 min, 15 min, 20 min, and 25 min,

each at 5-minute intervals. In this study, the intermediate sizes of the lag observations are selected by trial and error after noticing the negative impact of increasing the amount of input data. The assessment presented is conducted using the hyperparameters described in Table 2.

Table 2 lists the statistical outcomes of the deep learning algorithms—LSTM-GRU, LSTM, GRU-LSTM, and GRU—regarding 5-minute wind speed predictions across various input lag configurations (Lag-5, Lag-10, Lag-15, Lag-20, and Lag-25). The model hyperparameters are chosen using validation-driven empirical tuning following a fixed chronological train/validation/test methodology, ensuring that all designs are evaluated under uniform conditions. The provided metrics, comprising MSE, R^2, RMSE, nRMSE, MAE, and nMAE, are used to evaluate these models. Smaller values of MSE, RMSE, MAE, nRMSE, and nMAE show enhanced predictive accuracy, but higher R^2 values imply a more robust model fit to the data.

Table 2. Hyperparameters used for the deep learning algorithms (FC 1 = fully connected layer with 1 neuron for regression output).

Model	Layers & Units	Dropout	Optimizer	Learning Rate	Epoch
LSTM-GRU	LSTM 128 → GRU 64 → GRU 32 → FC 1	0.25 after each recurrent layer	Adam	0.001	100
LSTM	LSTM 128 → LSTM 64 → LSTM 32 → FC 1	0.25	Adam	0.001	100
GRU-LSTM	GRU 128 → LSTM 64 → LSTM 32 → FC 1	0.25 after each recurrent layer	Adam	0.001	100
GRU	GRU 128 → GRU 64 → GRU 32 → FC 1	0.25 after each recurrent layer	Adam	0.001	100

Table 3 demonstrates that the LSTM-GRU hybrid model consistently surpasses the performance of both standalone LSTM and GRU networks, in addition to the GRU-LSTM, across all statistical metrics. The LSTM-GRU model demonstrates the lowest average MSE (0.00120 m^2/s^2), RMSE (0.03431 m/s), MAE (0.02720 m/s), nRMSE (0.01034%), and nMAE (0.00820%), alongside the highest average R^2 (0.99932). The outstanding efficiency indicates that the combined use of LSTM and GRU units successfully captures the complex temporal dependencies and long-range correlations found in the preprocessed wind speed data. The capacity of LSTM to manage vanishing and exploding gradients, along with the efficiency and compact design of GRU, seems to result in a strong model that can achieve highly accurate predictions. Despite the attainment of elevated R^2 values, the risk of overfitting was alleviated by employing distinct validation/testing subsets, dropout-based regularization, and uniform out-of-sample assessment across all benchmark models.

Table 3. Statistical results of the proposed model for 5-minute forecasts with deep learning algorithms, including MSE (m^2/s^2), RMSE and MAE (m/s), and nRMSE and nMAE (%).

Algorithm	Input Data	MSE	R^2	RMSE	nRMSE	MAE	nMAE
LSTM-GRU	Lag-5	0.00081	0.99954	0.02840	0.00856	0.02244	0.00676
	Lag-10	0.00110	0.99937	0.03318	0.01000	0.02644	0.00797
	Lag-15	0.00087	0.99951	0.02944	0.00887	0.02340	0.00705
	Lag-20	0.00166	0.99906	0.04071	0.01226	0.03215	0.00968
	Lag-25	0.00158	0.99910	0.03981	0.01199	0.03159	0.00952
	Average	0.00120	0.99932	0.03431	0.01034	0.02720	0.00820
LSTM	Input Data	MSE	R^2	RMSE	nRMSE	MAE	nMAE
	Lag-5	0.00154	0.99913	0.03923	0.42636	0.02970	0.32285
	Lag-10	0.00343	0.99805	0.05859	0.63683	0.04322	0.46981
	Lag-15	0.00214	0.99879	0.04621	0.50224	0.03471	0.37731
	Lag-20	0.00273	0.99845	0.05221	0.56746	0.03830	0.41631
	Lag-25	0.00362	0.99794	0.06018	0.65408	0.04427	0.48121
	Average	0.00269	0.99847	0.05128	0.55740	0.03804	0.41350
GRU-LSTM	Input Data	MSE	R^2	RMSE	nRMSE	MAE	nMAE
	Lag-5	0.00146	0.99917	0.03817	0.41489	0.02797	0.30403
	Lag-10	0.00204	0.99884	0.04519	0.49121	0.03367	0.36603
	Lag-15	0.00209	0.99881	0.04574	0.49713	0.03360	0.36527
	Lag-20	0.00230	0.99869	0.04799	0.52168	0.03789	0.41180
	Lag-25	0.00225	0.99872	0.04742	0.51544	0.03670	0.39888
	Average	0.00322	0.99817	0.05393	0.58620	0.04084	0.44396
GRU	Input Data	MSE	R^2	RMSE	nRMSE	MAE	nMAE
	Lag-5	0.00189	0.99892	0.04350	0.47283	0.03297	0.35842
	Lag-10	0.00253	0.99856	0.05029	0.54661	0.03543	0.38513
	Lag-15	0.00543	0.99691	0.07372	0.80125	0.05994	0.65149
	Lag-20	0.00277	0.99842	0.05266	0.57238	0.04085	0.44403
	Lag-25	0.00773	0.99561	0.08789	0.95537	0.06812	0.74045
	Average	0.00407	0.99769	0.06161	0.66969	0.04746	0.51590

The LSTM-GRU model demonstrates optimal performance at Lag-5 and Lag-15, attaining the minimal MSE values of 0.00081 and 0.00087 m^2/s^2, respectively. This implies that the LSTM-GRU model requires a brief input history (5 or 15 time steps) to effectively identify meaningful patterns for 5-minute wind speed predictions. This can be attributed to the efficacy of the prior decomposition and clustering processes in capturing significant information. On the other hand, performance appears to diminish marginally with extended lags (Lag-20 and Lag-25), indicating that more historical data may introduce noise or redundant information, or that the model's efficacy is maximized with shorter sequences. The consistently elevated R^2 values exceeding 0.999 for LSTM-GRU across all lags further emphasize its robust prediction capacity and superior fit to the wind speed data.

Conversely, the standalone LSTM and GRU models, although exhibiting elevated R^2 values, present greater error metrics in comparison to LSTM-GRU. The LSTM model exhibits an average

MSE of 0.00269 m^2/s^2, which is over double that of LSTM-GRU. The GRU model demonstrates the greatest average error metrics among the deep learning models, with an average MSE of 0.00407 m^2/s^2. The GRU-LSTM combination has a lower efficiency compared to LSTM-GRU, with an average MSE of 0.00322 m^2/s^2. These findings highlight the importance of architectural decisions in deep learning models for time series forecasting, particularly when dealing with highly dynamic and complex datasets, such as wind speed. The precise configuration and interplay between LSTM and GRU layers in the LSTM-GRU setup appear to be especially effective in utilizing the enhanced features from the proposed preprocessing framework.

Furthermore, Figure 6 demonstrates the effectiveness of the proposed approach utilizing the LSTM-GRU forecasting model in comparison to GRU-LSTM, GRU, LSTM, ANN, SVM, and DT. Figure 6(a) illustrates that when the ideal input is correctly configured, the proposed LSTM-GRU model surpasses other models in properly tracking the actual wind speed output values. Additionally, the boxplots presented in Figure 6(b) aim to provide a more thorough evaluation of the predictive efficacy of the forecasting models. A box and whisker plot (BWP) illustrates the distribution of the MAE for all evaluated deep and machine learning techniques. In the analysis of the BWP, an outlier is a data point that quantitatively diverges from the other data. In accordance with previous conclusions, the proposed approach utilizing LSTM-GRU models regularly surpasses other deep learning and machine learning models.

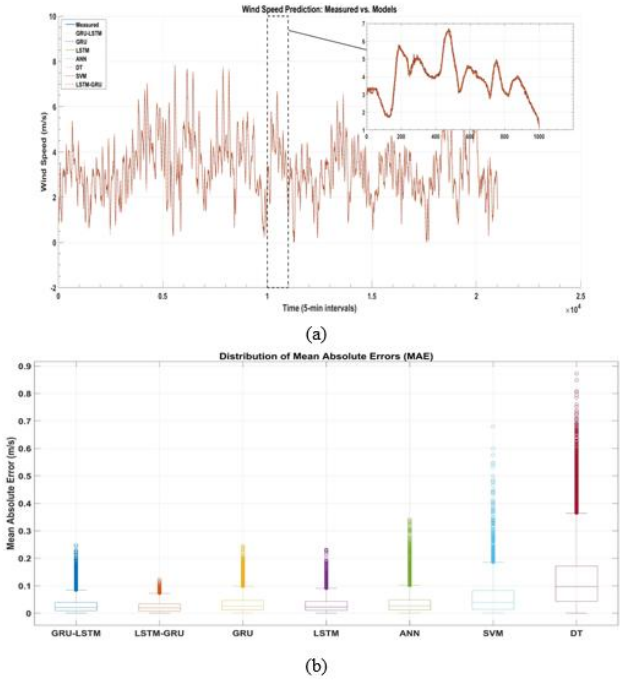


Figure 6. (a) The efficacy of the proposed method with the LSTM-GRU forecasting model in comparison to GRU-LSTM, GRU, LSTM, ANN, SVM, and DT; (b) boxplot comparing the MAE error values of all the considered deep and machine learning algorithms.

In addition, the exceptional performance is further illustrated in the scatter plots depicted in Figure 7. This chart illustrates the measured versus predicted wind speed output values obtained from the proposed LSTM-GRU model in comparison to other forecasting models evaluated in this study. In

addition, the convergence curves for Lag-5 and Lag-25, depicted in Figure 8, demonstrate that all models experience a fast initial decrease in loss, which stabilizes after the initial training epochs, signifying effective learning behavior. Hybrid architectures (GRU-LSTM and LSTM-GRU) typically demonstrate more rapid and smoother convergence than the independent GRU and LSTM models, accompanied by reduced validation loss trends.

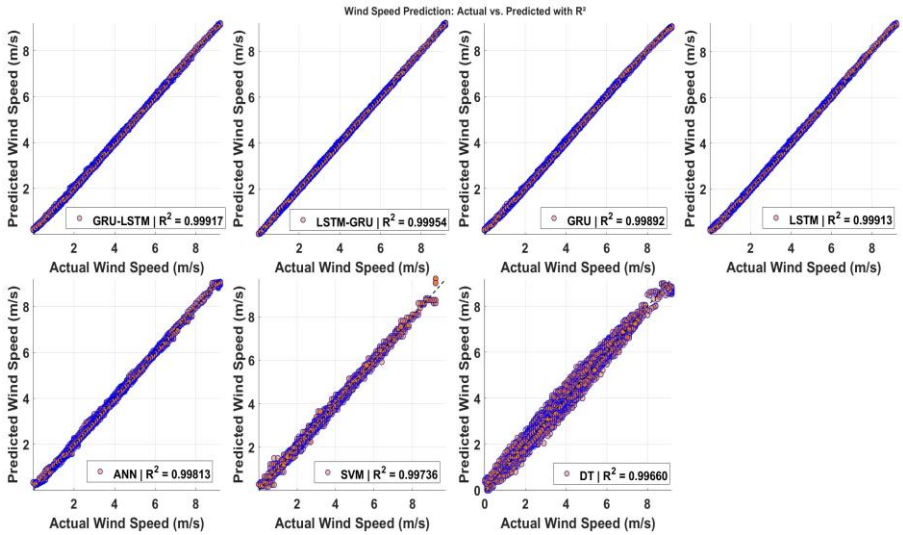
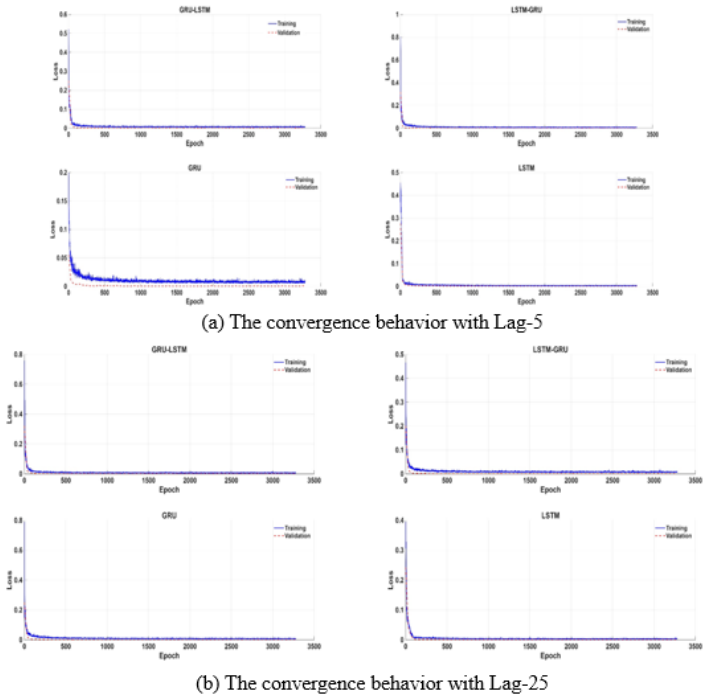


Figure 7. The measured vs. predicted wind speed output values acquired by the proposed LSTM-GRU model compared to other forecasting models considered in this study.



(a) The convergence behavior with Lag-5

(b) The convergence behavior with Lag-25

Figure 8. The convergence behavior of training and validation loss curves for various models with (a) Lag-5 and (b) Lag-25, demonstrating the learning stability and comparative efficacy of the GRU-LSTM, LSTM-GRU, GRU, and LSTM architectures.

3.2. Performance of conventional machine learning algorithms

The performance of the three conventional machine learning algorithms, namely, ANN, SVM, and DT, is investigated using the refined IMFs generated by the proposed method. The hyperparameters and architectural setups for these models are chosen to prevent overfitting. The SVM uses a linear kernel function to project the features into a higher-dimensional space. The soft-margin parameter, C, of the linear kernel function is determined by an empirical trial-and-error method to identify the value that leads to the best accuracy value and generalization performance. The ANN is constructed utilizing a multilayer perceptron (MLP) and the backpropagation algorithm, employing the Levenberg–Marquardt technique as the training function. Three layers are identified in the back propagation neural network (BPNN): input, hidden, and output. The quantity of neurons in the hidden layer is determined to be equivalent to the size of the lag observations of input data. For example, 10 neurons are chosen with the input data that contains Lag-10 observations of the refined IMFs. The ANN utilizes dropout (rate 0.3) and L2 weight regularization ($\lambda = 0.001$) to mitigate neuronal co-adaptation and to penalize excessive weights. The DT had 100 decision trees trained using bootstrap aggregation and Gini impurity for split optimization. Table 4 displays the statistical findings for the conventional machine learning algorithms: ANN, SVM, and DT. The models are assessed for 5-minute wind speed predictions using the same input lag configurations as the deep learning models.

Table 4. Statistical results of the proposed model for 5-minute forecasts with conventional machine learning algorithms, including MSE (m^2/s^2), RMSE and MAE (m/s), and nRMSE and nMAE (%).

Algorithm	Input Data	MSE	R^2	RMSE	nRMSE	MAE	nMAE
ANN	Lag-5	0.00328	0.99813	0.05729	0.01726	0.04093	0.01233
	Lag-10	0.00356	0.99798	0.05968	0.01798	0.04202	0.01266
	Lag-15	0.00329	0.99813	0.05733	0.01727	0.04101	0.01235
	Lag-20	0.00329	0.99813	0.05733	0.01727	0.04110	0.01238
	Lag-25	0.00335	0.99810	0.05785	0.01743	0.04151	0.01250
	Average	0.00337	0.99808	0.05805	0.01749	0.04141	0.01247
SVM	Input Data	MSE	R^2	RMSE	nRMSE	MAE	nMAE
	Lag-5	0.00464	0.99736	0.06808	0.02051	0.04872	0.01468
	Lag-10	0.00522	0.99703	0.07228	0.02177	0.05089	0.01533
	Lag-15	0.00504	0.99713	0.07100	0.02139	0.04978	0.01500
	Lag-20	0.00486	0.99724	0.06973	0.02101	0.04909	0.01479
	Lag-25	0.00512	0.99709	0.07158	0.02157	0.05028	0.01515
	Average	0.00506	0.99712	0.07115	0.02143	0.05001	0.01507
DT	Input Data	MSE	R^2	RMSE	nRMSE	MAE	nMAE
	Lag-5	0.00598	0.99660	0.07730	0.02329	0.05489	0.01654
	Lag-10	0.00606	0.99655	0.07788	0.02346	0.05521	0.01663
	Lag-15	0.00653	0.99629	0.08084	0.02435	0.05603	0.01688
	Lag-20	0.00689	0.99608	0.08300	0.02500	0.05710	0.01720
	Lag-25	0.00714	0.99594	0.08452	0.02546	0.05756	0.01734
	Average	0.00652	0.99629	0.08071	0.02431	0.05616	0.01692

As shown in Table 4, among conventional machine learning techniques, the ANN exhibits superior performance, attaining the lowest average MSE (0.00337 m^2/s^2), RMSE (0.05805 m/s), MAE (0.04141 m/s), nRMSE (0.01749%), and nMAE (0.01247%), alongside an average R^2 of 0.99808. The performance of the ANN remains consistently stable across various lag configurations, demonstrating its robustness in managing the preprocessed features. This indicates that the ANN, albeit architecturally less complex than RNNs, can still proficiently learn from the complex patterns derived from the VMD-PermEn-DBSCAN-ICEEMDAN method.

The SVM exhibits a marginally elevated average error relative to the ANN, with an average MSE of 0.00506 m^2/s^2. Although SVMs are recognized for their robust generalization ability, especially in high-dimensional spaces, their performance in this study suggests that the non-linear mapping proficiency of ANNs may be more beneficial for this particular time series forecasting objective. The DT demonstrates the highest error metrics among all examined models, with an average MSE of 0.00652 m^2/s^2. This is expected, given DTs generally demonstrate reduced capability in capturing complex temporal dependencies and non-linear interactions in time series data compared to ANNs or SVMs, even with effective preprocessing methods. Their performance is more susceptible to the inherent unpredictability and non-stationarity of wind speed data.

A comparison of the deep learning models (Table 3) with the standard machine learning models (Table 4) reveals a distinct trend: the deep learning models, especially LSTM-GRU, markedly surpass their traditional equivalents. The MSE of LSTM-GRU (0.00120 m^2/s^2) is significantly lower than that of the ANN (0.00337 m^2/s^2), SVM (0.00506 m^2/s^2), and DT (0.00652 m^2/s^2). This highlights the enhanced capacity of deep learning architectures, particularly those designed for sequential data, to identify and integrate complex patterns from thoroughly processed wind speed components. The hybrid decomposition and feature engineering processes are essential for both model categories; however, deep learning models are more adept at utilizing the extensive information included in the input dataset.

3.3. Comparison of input data

The predicting accuracy of both deep learning and conventional machine learning models, as presented in Tables 3 and 4, is significantly affected by the selection of the input lag length. According to Tables 3 and 4, Lag-5 consistently yields the optimal accuracy across all models. For deep learning models, the LSTM-GRU attains its optimal error at Lag-5, with MSE = 0.00081 m^2/s^2, RMSE = 0.0284 m/s, and nRMSE = 0.00856%. However, extending the lag to 25 results in performance deterioration, with MSE = 0.00158 m^2/s^2 and RMSE = 0.03981 m/s, indicating an error escalation of about 40%. A similar tendency is noted for LSTM, with RMSE escalating from 0.03923 m/s (Lag-5) to 0.06018 m/s (Lag-25), and for GRU, where RMSE markedly increases from 0.04350 m/s (Lag-5) to 0.08789 m/s (Lag-25). In conventional models, the ANN demonstrates its minimal RMSE at Lag-5 (0.05729 m/s), which escalates to 0.05785 m/s at Lag-25. Similarly, both the SVM and DT display a steady increase in error as the lag observations are increased. For the SVM, the RMSE rises from 0.06808 m/s (Lag-5) to 0.07158 m/s (Lag-25), and for the DT from 0.07730 m/s to 0.08452 m/s. Therefore, the optimal lag (Lag-5) is employed in the subsequent comparisons with hybrid models (Subsection 3.4) and in sensitivity assessments across various forecasting horizons (Subsection 3.5).

3.4. Comparison with different hybrid models

Table 5 presents a key comparison of the proposed VMD-PermEn-DBSCAN-ICEEMDAN hybrid model with existing simplified hybrid methodologies, namely, VMD-LSTM-GRU and ICEEMDAN-LSTM-GRU. This comparison is essential for comprehending the incremental worth of each element within the overall framework.

Table 5. Statistical results of the proposed model for 5-minute forecasts compared with other models, including MSE (m^2/s^2), RMSE and MAE (m/s), and nRMSE and nMAE (%).

Model	MSE	R^2	RMSE	nRMSE	MAE	nMAE
VMD-LSTM-GRU	0.00583	0.99669	0.07637	0.83008	0.06099	0.66290
ICEEMDAN-LSTM-GRU	0.00407	0.99769	0.06382	0.69374	0.04623	0.50250
VMD-PermEn-DBSCAN-ICEEMDAN	0.00081	0.99954	0.02840	0.00856	0.02244	0.00676

Table 5 clearly illustrates the enhanced efficacy of the proposed VMD-PermEn-DBSCAN-ICEEMDAN hybrid model. The proposed comprehensive framework greatly surpasses both the VMD-LSTM-GRU and ICEEMDAN-LSTM-GRU models, evidenced by an MSE of 0.00081 m^2/s^2, an R^2 of 0.99954, and low error metrics (RMSE: 0.02840 m/s, nRMSE: 0.00856%, MAE: 0.02244 m/s, nMAE: 0.00676%). The VMD-LSTM-GRU model that employs VMD for decomposition and then LSTM-GRU for prediction, omitting the intermediate steps of PermEn, DBSCAN clustering, and targeted ICEEMDAN refinement, exhibits a significantly elevated MSE of 0.00583 m^2/s^2. Likewise, the ICEEMDAN-LSTM-GRU model, which applies ICEEMDAN directly on the raw signal prior to inputting it into LSTM-GRU, demonstrates elevated error metrics (MSE: 0.00407 m^2/s^2) in comparison to the proposed model.

This comparative evaluation highlights the essential role of every stage in the proposed hybrid wind speed prediction model. The incorporation of PermEn for feature extraction and DBSCAN for intelligent clustering facilitates enhanced categorization and segregation of various signal components. Moreover, the precise application of ICEEMDAN to high-frequency clusters allows a more detailed decomposition, identifying subtle patterns essential for reliable short-term forecasting. These intermediate processes combined yield a more refined and useful input for the deep learning prediction models, significantly decreasing prediction errors. The findings support the idea that a multi-stage, adaptive preprocessing method, customized for wind speed data, is crucial for attaining superior forecasting performance. The synergistic integration of VMD, PermEn, DBSCAN, and ICEEMDAN establishes a robust data preparation pipeline that markedly improves the prediction efficacy of sophisticated deep learning algorithms such as LSTM-GRU.

3.5. Performance across different forecasting resolutions

For further analysis, Table 6 displays the statistical outcomes of the proposed model over several predicting intervals: 15 min, 30 min, 45 min, and 1 hr. This analysis is essential for comprehending the model's applicability and robustness across diverse prediction horizons, which is beneficial for various operational and planning dimensions of wind energy systems.

As shown in Table 6, the model's prediction accuracy decreases as the forecasting resolution, or prediction horizon, increases. For a 15-minute forecast, the model exhibits a high degree of accuracy,

evidenced by an MSE of 0.00321 m^2/s^2 and an R^2 of 0.99817. This result supports the above findings, demonstrating the model's robust capacity for ultra-short-term forecasts. Nevertheless, as the resolution increases to 30 min, 45 min, and 1 hr, there is a significant rise in error metrics (MSE, RMSE, MAE, nRMSE, nMAE) with a continued reduction in R^2. For instance, at the 1-hour resolution, the MSE increases to 0.05206 m^2/s^2, while the R^2 decreases to 0.97007. This pattern aligns with the inherent difficulties of time series forecasting, wherein uncertainty compounds over extended prediction horizons due to the chaotic and stochastic nature of atmospheric phenomena. The model's capacity to maintain an R^2 over 0.97 for 1-hour forecasts exhibits the efficacy of the VMD-PermEn-DBSCAN-ICEEMDAN model in deriving stable and predictable elements from the wind speed signal.

Table 6. Statistical results of the proposed model with different forecasting resolutions, including MSE (m^2/s^2), RMSE and MAE (m/s), and nRMSE and nMAE (%).

Resolution	MSE	R^2	RMSE	nRMSE	MAE	nMAE
15 min	0.00321	0.99817	0.05665	0.01707	0.04107	0.01237
30 min	0.01010	0.99424	0.10049	0.03027	0.07256	0.02186
45 min	0.02961	0.98306	0.17208	0.05183	0.12123	0.03652
1 hour	0.05206	0.97007	0.22817	0.06872	0.16439	0.04951

Although errors increase with extended horizons, the model continues to yield valuable forecasts, which are essential for operational planning and grid management at these resolutions. The findings also indicate that the proposed framework is especially adept at ultra-short-term wind speed forecasting (e.g., 5 to 15 min), since its extensive signal processing capabilities produce remarkable accuracy. For longer horizons, although accuracy declines, the model maintains a high prediction accuracy, surpassing numerous traditional methods that show low prediction outcomes due to the variability of wind speed data. To further visualize this finding, Figure 9 (left) shows the performance of the proposed method with LSTM-GRU when the actual wind speed data is plotted against predicted values. Figure 9 (right) also illustrates the MAE over various forecasting resolutions, demonstrating a consistent decrease in the accuracy of forecasting models, beginning with the optimal prediction at 15 min and finishing at a 1 hr forecast. As the time horizon extends, the precision of various models progressively diminishes, and the uncertainty in wind speed predictions increases.

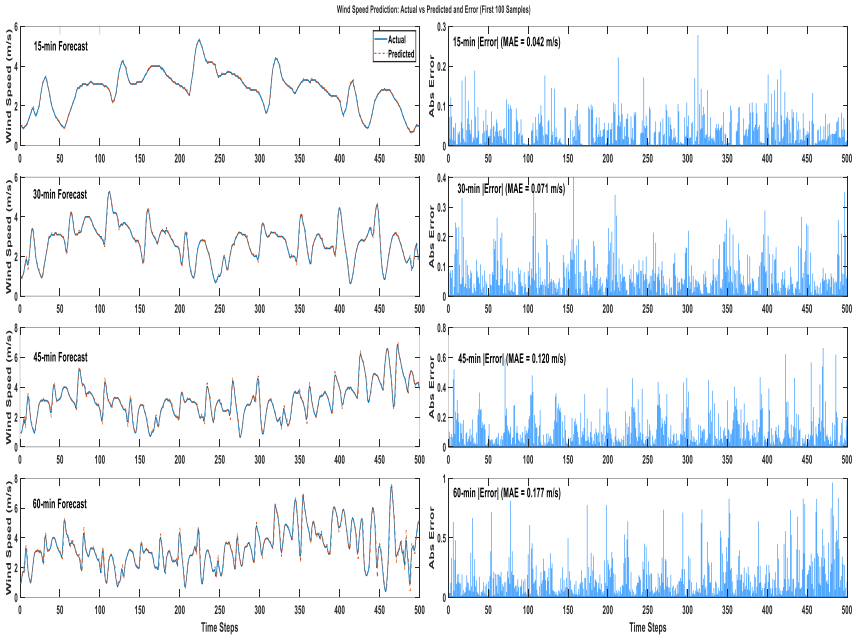


Figure 9. (Left) The measured vs. predicted wind speed output values acquired by the proposed LSTM-GRU model with various forecasting resolutions; (Right) MAE over various forecasting resolutions.

4. Conclusions

This study developed and evaluated a hybrid framework for ultra-short-term wind speed forecasting, combining advanced signal processing methods with deep learning models. The proposed methodology attempted to tackle the inherent problems of non-linearity and non-stationarity in wind speed data by integrating variational mode decomposition (VMD), permutation entropy (PermEn), DBSCAN clustering, and ICEEMDAN. The multi-stage refining procedure guaranteed that the forecasting models received high-quality, feature-rich inputs, resulting in markedly enhanced predictive performance. The examination of different deep learning and conventional machine learning models revealed that the combined use of these methods is essential for accurately capturing the intricate dynamics of wind speed data.

The findings highlighted the dominance of the LSTM-GRU hybrid model inside the proposed framework, which consistently attained the highest level of precision across several statistical criteria. The comparative analysis against conventional models and simplified hybrid approaches demonstrated that every phase of the preprocessing pipeline significantly contributes to minimizing forecasting error. Although accuracy diminishes as the forecasting period lengthens from 5 minutes to 1 hour, the model exhibited strong performance, demonstrating its effectiveness for real-time grid management and operational planning. The main outputs of this study are summarized as follows:

- The VMD-PermEn-DBSCAN-ICEEMDAN-LSTM-GRU framework demonstrated superior predictive accuracy, attaining an MSE of 0.00081 m/s and an R^2 value surpassing 0.999 for ultra-short-term forecasts.
- The LSTM-GRU configuration demonstrated superior performance compared to individual LSTM and GRU models, as well as conventional techniques for machine learning such as ANN, SVM,

and DT, by effectively leveraging the refined temporal features.

- The incorporation of DBSCAN clustering and focused ICEEMDAN refining markedly improved the model's capacity to manage high-frequency fluctuations relative to conventional decomposition techniques.
- The study revealed that a reduced input lag (Lag-5) produces the greatest accuracy, as a large amount of past data may generate noise that hinders the prediction of highly dynamic wind signals.
- The framework exhibited consistent performance across many forecasting horizons, sustaining a high explanatory power ($R^2 > 0.97$) even at one-hour intervals, which is crucial for practical wind energy applications.

This study introduces a resilient and successful hybrid framework for ultra-short-term wind speed forecasting; nonetheless, significant shortcomings require examination and provide opportunities for further research. For example, this research utilized wind speed data from a particular geographical location, Riyadh, Saudi Arabia. Although the framework has shown high accuracy results, its applicability to various climatic environments and geographical terrains, which may display distinct wind speed characteristics, has yet to be thoroughly investigated. Further studies can investigate the performance of the framework over various locations to evaluate its flexibility and resilience.

In addition, based on the outcomes of this research, several potential avenues for future research can expand upon the fundamental work offered here. One potential area is the optimization of hyperparameter tuning for deep learning models and the preprocessing phases. The present work utilized a trial-and-error methodology for specific parameters, such as lag observations; however, the incorporation of advanced optimization techniques (e.g., genetic algorithms, Bayesian optimization) may enhance performance and yield more generic designs. Another promising direction is the integration of real-time external data sources, such as numerical weather prediction (NWP) outputs, to enhance the input features, especially for longer forecasting horizons. In addition, researchers can expand this study by incorporating and benchmarking recent transformer-based forecasting architectures such as the informer, autoformer, and PatchTST to further evaluate and enhance model performance. Finally, the overall computing burden of the framework is manageable for real-time execution, as each decomposition and refinement phase operates on a univariate series or a limited set of IMFs. The ultimate prediction models are streamlined recurrent networks suitable for deployment on contemporary embedded platforms with limited memory requirements.

Use of Generative-AI tools declaration

The author declares he has not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

The researcher would like to thank the Deanship of Graduate Studies and Scientific Research at Qassim University for the financial support (QU-APC-2026).

Conflict of interest

The author declares that he has no competing interests.

References

[1] J. Selvaraj, L. Muthuramalingam, V. Karthikeyan, A. Karthick, V. Sathiyaseelan, Optimizing wind energy integration: A review of forecasting techniques and emerging trends, *Arch. Computat. Methods Eng.*, **33** (2026), 4261–4286. https://doi.org/10.1007/s11831-025-10442-1

[2] D. Kaur, T. T. Lie, N. K. C. Nair, B. Vallès, Wind speed forecasting using hybrid wavelet transform—ARMA techniques, *AIMS Energy*, **3** (2015), 13–24. https://doi.org/10.3934/energy.2015.1.13

[3] Y. Zhou, S. P. Guan, Y. W. Xue, Hierarchical multi-scale temporal-frequency pattern extraction for accurate wind speed forecasting, *Pattern Anal. Applic.*, **29** (2026), 26. https://doi.org/10.1007/s10044-025-01606-7

[4] S. Ansay, B. Köse, İ. Işıklı, C. Mülayim, B. Ertilav, Forecasting wind speed with autoregressive and long-short term memory neural network models, *Journal of AI*, **9** (2025) 98–121.

[5] Y. R. Wang, D. C. Wang, Y. Tang, Clustered hybrid wind power prediction model based on ARMA, PSO-SVM, and clustering methods, *IEEE Access*, **8** (2020), 17071–17079. https://doi.org/10.1109/ACCESS.2020.2968390

[6] E. Cadenas, W. Rivera, R. Campos-Amezcua, C. Heard, Wind speed prediction using a univariate ARIMA model and a multivariate NARX model, *Energies*, **9** (2016), 109. https://doi.org/10.3390/en9020109

[7] W. Q. Xu, L. K. Ning, Y. Luo, Wind speed forecast based on post-processing of numerical weather predictions using a gradient boosting decision tree algorithm, *Atmosphere*, **11** (2020), 738. https://doi.org/10.3390/atmos11070738

[8] J. Du, S. Z. Chen, L. L. Pan, Y. B. Liu, A wind speed prediction method based on signal decomposition technology deep learning model, *Energies*, **18** (2025), 1136. https://doi.org/10.3390/en18051136

[9] A. Kumar, A. J. Singh, S. Kumar, Forecasting wind speed over multiple horizons: Superiority of deep learning and decomposition-driven hybrid models, *Unconventional Resources*, **9** (2026), 100296. https://doi.org/10.1016/j.uncres.2025.100296

[10] M. Alrashidi, Ultra-short-term solar forecasting with reduced pre-acquired data considering optimal heuristic configurations of deep neural networks, *AIMS Mathematics*, **9** (2024), 12323–12356. https://doi.org/10.3934/math.2024603

[11] P. Pinson, C. Chevallier, G. N. Kariniotakis, Trading wind generation from short-term probabilistic forecasts of wind power, *IEEE T. Power Syst.*, **22** (2007), 1148–1156. https://doi.org/10.1109/TPWRS.2007.901117

[12] S. S. Soman, H. Zareipour, O. Malik, P. Mandal, A review of wind power and wind speed forecasting methods with different time horizons, In: *North American Power Symposium 2010*, Arlington, TX, USA, 2010, 1–8. https://doi.org/10.1109/NAPS.2010.5619586

[13] F. Li, H. Z. Wang, D. Wang, D. Liu, K. Sun, A review of wind pwer prediction methods based on multi-time scales, *Energies*, **18** (2025), 1713. https://doi.org/10.3390/en18071713

[14] F. L. Dias, A. J. Naik, A novel approach to wavelet neural network-based wind power forecasting, *Wind*, **5** (2025), 14. https://doi.org/10.3390/wind5020014

[15] A. E. Kio, J. Xu, N. Gautam, Y. Ding, Wavelet decomposition and neural networks: a potent combination for short term wind speed and power forecasting, *Front. Energy Res.*, **12** (2024), 1277464. https://doi.org/10.3389/fenrg.2024.1277464

[16] G. N. Pillai, K. V. Shihabudheen, Wind speed forecasting using empirical mode decomposition and regularized ELANFIS, In: *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, Honolulu, HI, USA, 2017, 1–7. https://doi.org/10.1109/SSCI.2017.8285324

[17] Y. Z. Du, K. Zhang, Q. Z. Shao, Z. Chen, A short-term prediction model of wind power with outliers: An integration of long short-term memory, ensemble empirical mode decomposition, and sample entropy, *Sustainability*, **15** (2023), 6285. https://doi.org/10.3390/su15076285

[18] Z. Xing, Y. Zhi, R. H. Hao, H. W. Yan, C. Qing, Wind speed forecasting model based on extreme learning machines and complete ensemble empirical mode decomposition, In: *2020 5th Asia Conference on Power and Electrical Engineering (ACPEE)*, Chengdu, China, 2020, 159–163. https://doi.org/10.1109/ACPEE48638.2020.9136553

[19] B. S. Bommidi, V. Kosana, K. Teeparthi, S. Madasthu, A hybrid approach to ultra short-Term wind speed prediction using CEEMDAN and informer, In: *2022 22nd National Power Systems Conference (Npsc)*, New Delhi, India, 2022, 207–212. https://doi.org/10.1109/NPSC57038.2022.10069064

[20] K. Dragomiretskiy, D. Zosso, Variational mode decomposition, *IEEE T. Signal Proces.*, **62** (2014), 531–544. https://doi.org/10.1109/TSP.2013.2288675

[21] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Snin, Q. A. Zheng, et al., The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis, *Proc. A*, **454** (1998), 903–995. https://doi.org/10.1098/rspa.1998.0193

[22] X. X. Wang, P. Liu, Q. Y. Liu, Y. Z. Zhang, A hybrid ultra-short-term wind speed prediction model using adaptive VMD and time-series mixer, *Int. J. Green Energy*, **22** (2025), 2794–2812. https://doi.org/10.1080/15435075.2025.2471997

[23] J. He, Z. J. Cheng, Z. J. Zhong, L. Z. Liang, J. H. Ye, Wind power prediction based on a hybrid model of iceemdan and moderntcn-informer, *IEEE Access*, **13** (2025), 145256–145270. https://doi.org/10.1109/ACCESS.2025.3599609

[24] W. Li, C. X. Quan, X. Y. Wang, S. Zhang, Short-term power load forecasting based on a combination of VMD and ELM, *Pol. J. Environ. Stud.*, **27** (2018), 2143–2154. https://doi.org/10.15244/pjoes/78244

[25] S. J. Wang, C. Liu, K. Liang, Z. Y. Cheng, X. Kong, S. Gao, Wind speed prediction model based on improved vmd and sudden change of wind speed, *Sustainability*, **14** (2022), 8705. https://doi.org/10.3390/su14148705

[26] X. X. Wang, X. P. Shen, X. Y. Ai, S. J. Li, Short-term wind speed forecasting based on a hybrid model of ICEEMDAN, MFE, LSTM and informer, *PLoS One*, **18** (2023), e0289161. https://doi.org/10.1371/journal.pone.0289161

[27] Z. R. Shu, H. C. Deng, P. W. Chan, X. H. He, Evaluating the intrinsic predictability of wind speed time series via entropy-based approaches, *J. Wind Eng. Ind. Aerod.*, **257** (2025), 105972. https://doi.org/10.1016/j.jweia.2024.105972

[28] J. J. Ruiz-Aguilar, I. Turias, J. González-Enrique, D. Urda, D. Elizondo, A permutation entropy-based EMD–ANN forecasting ensemble approach for wind speed prediction, *Neural Comput. Applic.*, **33** (2021), 2369–2391. https://doi.org/10.1007/s00521-020-05141-w

[29] A. Azhar, H. Hashim, A review of wind clustering methods based on the wind speed and trend in Malaysia, *Energies*, **16** (2023), 3388. https://doi.org/10.3390/en16083388

[30] P. Zhang, Y. L. Wang, L. K. Liang, X. Li, Q. T. Duan, Short-term wind power prediction using GA-BP neural network based on DBSCAN algorithm outlier identification, *Processes*, **8** (2020), 157. https://doi.org/10.3390/pr8020157

[31] S. Q. Wang, C. Chen, Short-term wind power prediction based on DBSCAN clustering and support vector machine regression, In: *2020 5th International Conference on Computer and Communication Systems (ICCCS)*, Shanghai, China, 2020, 941–945. https://doi.org/10.1109/ICCCS49078.2020.9118606

[32] H. Liu, C. Chen, Data processing strategies in wind energy forecasting models and applications: A comprehensive review, *Appl. Energ.*, **249** (2019), 392–408. https://doi.org/10.1016/j.apenergy.2019.04.188

[33] A. Taha, N. Nazih, P. Makeen, Wind speed prediction based on variational mode decomposition and advanced machine learning models in zaafarana, Egypt, *Sci. Rep.*, **15** (2025), 15599. https://doi.org/10.1038/s41598-025-98543-6

[34] H. L. Hu, L. Wang, R. Tao, Wind speed forecasting based on variational mode decomposition and improved echo state network, *Renew. Energ.,* **164** (2021), 729–751. https://doi.org/10.1016/j.renene.2020.09.109

[35] C. Bandt, B. Pompe, Permutation entropy: A natural complexity measure for time series, *Phys. Rev. Lett.*, **88** (2002), 174102. https://doi.org/10.1103/PhysRevLett.88.174102

[36] D. T. Mihailović, Permutation entropy and its niche in hydrology: A review, *Entropy*, **27** (2025), 598. https://doi.org/10.3390/e27060598

[37] Y. Sun, S. Y. Zhang, A multiscale hybrid wind power prediction model based on least squares support vector regression-regularized extreme learning machine-multi-head attention-bidirectional gated recurrent unit and data decomposition, *Energies*, **17** (2024), 2923. https://doi.org/10.3390/en17122923

[38] Z. L. Zhang, Z. T. Xiang, Y. F. Chen, J. Y. Xu, Fuzzy permutation entropy derived from a novel distance between segments of time series, *AIMS Mathematics*, **5** (2020), 6244–6260. https://doi.org/10.3934/math.2020402

[39] M. Ester, H.-P. Kriegel, J. Sander, X. W. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, In: *Second International Conference on Knowledge Discovery and Data Mining (KDD'96)*, 1996, 226–231.

[40] M. A. Colominas, G. Schlotthauer, M. E. Torres, Improved complete ensemble EMD: A suitable tool for biomedical signal processing, *Biomed. Signal Proces.*, **14** (2014), 19–29. https://doi.org/10.1016/J.BSPC.2014.06.009

[41] B. J. Lu, Z. J. Li, X. B. Zhang, ICEEMDAN–RPE–AITD algorithm for magnetic field signals of magnetic targets, *Sci. Rep.*, **15** (2025), 6509. https://doi.org/10.1038/s41598-025-91068-y

[42] Q. Z. Zhuang, L. Gao, F. Zhang, X. Y. Ren, L. Qin, Y. P. Wang, MIVNDN: ultra-short-term wind power prediction method with MSDBO-ICEEMDAN-VMD-Nons-DC transformer net, *Electronics*, **13** (2024), 4829. https://doi.org/10.3390/electronics13234829

[43] M. Ayaz, T. Ali, M. Hijji, I. Baig, T. Alhmiedat, E. H. M. Aggoune, Predictive modeling of complex networks using deep learning and fractional dynamics, *AIMS Mathematics*, **10** (2025), 26717–26743. https://doi.org/10.3934/math.20251175

[44] Z. K. Luo, Learning dynamics and convergence of machine learning driven neural ordinary differential equations model for housing price prediction, *AIMS Mathematics*, **10** (2025), 23803–23820. https://doi.org/10.3934/math.20251058

[45] T. Zhang, An introduction to support vector machines and other kernel-based learning methods, *AI Mag.*, **22** (2001), 103–104.
[46] K.B. Sahay, S. Srivastava, Short- term wind speed forecasting of lelystad wind farm by using ANN algorithms, In: *2018 International Electrical Engineering Congress (IEECON)*, Krabi, Thailand, 2018, 1–4. https://doi.org/10.1109/IEECON.2018.8712227
[47] U. B. Kahveci, B. Barutçu, Recent hybrid machine learning approaches in wind speed forecasting: A review, *J. Econ. Surv.*, **40** (2025), 242–268. https://doi.org/10.1111/JOES.70005
[48] J. F. Ni, S. Y. Yang, Y. J. Liu, Data cleaning model of mine wind speed sensor based on LOF-GMM and SGAIN, *Appl. Sci.*, **15** (2025), 1801. https://doi.org/10.3390/app15041801
[49] R. Al-Hajj, Probabilistic machine learning-based forecasting of wind speed uncertainty using adaptive kernel density estimation, *Math. Biosci. Eng.*, **22** (2025), 2269–2299. https://doi.org/10.3934/mbe.2025083
[50] J. F. Lv, Y. X. Ren, H. J. Zhang, M. Han, A meta-inspired algorithm-based fuzzy model for wind speed prediction using chaotic time series and phase space reconstruction, *AIMS Energy*, **14** (2026), 1–22. https://doi.org/10.3934/energy.2026001