Results in Engineering 28 (2025) 107646

Contents lists available at ScienceDirect

Results in Engineering

journal homepage: www.sciencedirect.com/journal/results-in-engineering

Research paper

Hybrid model to improve wind energy prediction considering data granularity

Leslie Ricardo de la Rosa [a], Lía García-Pérez [b,*], Matilde Santos Peñas [c], Alejandro Gómez [d]

[a] Facultad de C.C. Físicas, Universidad Complutense, Plaza de las Ciencias 1, 28040, Madrid, Spain
[b] Dpto. de Arquitectura de Computadores y Automática, Facultad de C.C. Físicas, Universidad Complutense, Plaza de las Ciencias 1, 28040, Madrid, Spain
[c] Instituto de Ingeniería del Conocimiento, Facultad de Informática, Universidad Complutense, Calle del Prof. José García Santesmases, 9, 28040, Madrid, Spain
[d] Independent author, Madrid, Spain

ARTICLE INFO

Keywords:
Wind energy forecasting
Renewable energy
Machine learning
SARIMAX
XGBoost
LSTM
Hybrid model

ABSTRACT

The growth of wind energy generation as a renewable source in the transition to sustainable energy poses significant challenges in ensuring reliable production forecasting due to the intermittent nature of wind resources. The implementation of advanced forecasting models has become a priority to optimize its integration into the power grid and ensure the stability of the energy supply. This study focuses on improving wind energy predictions through the use of advanced machine learning techniques. The methodology includes a detailed analysis of different forecasting time horizons, sampling rates, and exogenous variable configurations, comparing traditional models such as SARIMAX, XGBoost, and LSTM neural networks with hybrid approaches. Furthermore, performance metrics such as R^2, MAE, RMSE and MAPE are evaluated to assess the accuracy and reliability of the proposed models. The results demonstrate that the selection of the optimal model depends on the forecasting horizon, data granularity, and available resources, maximizing precision and efficiency for each scenario.

1. Introduction

In response to climate change and the urgent need to reduce greenhouse gas emissions, the transition toward a sustainable energy model is advancing by leaps and bounds. In this context, wind power has become firmly established as one of the most promising renewable sources, due to its ability to produce clean and efficient electricity [1]. Generated by wind turbines that convert the kinetic energy of the wind into electrical energy, wind power has shown remarkable growth in both onshore and offshore installations, supplying large populations and reducing dependence on fossil fuels. However, the large-scale integration of wind energy into modern electrical grids poses challenges related to the inherent variability of wind and the need to ensure the stability and reliability of the power supply.

Technological advances and policies incentivizing carbon neutrality [2] have driven the rapid incorporation of wind farms into the energy mix. Nevertheless, the intermittent and sometimes unpredictable nature of wind makes balancing energy supply and demand more difficult [3]. Hence the importance of accurate short- and medium-term wind power forecasts for improving resource planning, grid operation, and participation in energy markets [4]. A reliable forecast can help reduce operational costs associated with activating backup reserves, energy storage, and demand management. Likewise, minimizing imbalances in energy delivery helps avoid financial penalties and strengthens grid stability, paving the way for greater integration of renewable sources and the gradual reduction of fossil fuels [5–7].

In this context, machine learning (ML) provides powerful tools to address these challenges, as the analysis of historical data makes it possible to identify patterns in wind behavior that are difficult to detect using traditional methods, allowing more accurate wind power forecasts. Furthermore, ML is applied in other areas, such as early fault detection in wind turbines by analyzing vibration and sensor data, optimizing predictive maintenance to minimize downtime, and improving energy storage management [8], thus contributing to maintaining operational efficiency without compromising the system's sustainability or stability. Despite these advances, most studies have focused on data improvements (for example, through filtering or imputing missing values) and on developing increasingly complex ML architectures, leaving rel-

* Corresponding author.
E-mail address: liagar05@ucm.es (L. García-Pérez).

https://doi.org/10.1016/j.rineng.2025.107646
Received 11 July 2025; Received in revised form 20 September 2025; Accepted 8 October 2025

Available online 10 October 2025

atively unexplored the effect of sampling frequency on forecast model performance. This factor is crucial because data granularity can affect both the quality of captured patterns and the computational complexity required to process them [9]. A sampling interval that is too large could overlook significant wind fluctuations, whereas one that is too short could overload the system with redundant information and increase computing time.

To address these limitations, this work makes two main contributions. First, it proposes a hybrid model that combines a noise filter, an LSTM (Long Short-Term Memory) network—a type of recurrent neural network designed to retain relevant information over extended sequences and mitigate the vanishing gradient problem—and the XGBoost (eXtreme Gradient Boosting) algorithm, a tree-based machine learning method. The filter reduces the effect of spurious data, the LSTM captures long-range temporal patterns, and XGBoost provides the final optimization of the prediction, leveraging the features identified in earlier stages. Second, it rigorously examines the role of data granularity in forecast accuracy by evaluating different sampling frequencies to determine their impact on model performance.

To validate the effectiveness of the proposed approach, the results of the hybrid model are compared to two methods widely employed in industry and research: SARIMAX (Seasonal AutoRegressive Integrated Moving Average with eXogenous variables), which accounts for trends and seasonality, and XGBoost (eXtreme Gradient Boosting). Additionally, the influence of exogenous variables—particularly wind speed and the power curve—on forecast accuracy is investigated, and the computational time required by each model is analyzed, a critical factor in determining the feasibility of real-world applications where both computing speed and predictive reliability are essential.

The overall structure of this work begins with a review of recent approaches in the field of wind energy forecasting, highlighting the main methodologies and the areas still open for exploration. This is followed by a description of the methodological framework and resources used, detailing both the selected machine learning models and the origin and characteristics of the datasets. Subsequently, the preprocessing and exploratory data analysis steps are presented, emphasizing the strategies used to enhance data quality. The main body of the study discusses the practical implementation of the proposed models, addressing issues such as temporal granularity, the inclusion of exogenous variables, and computational cost, and then presents and analyzes the results obtained. Finally, the conclusions summarize the key contributions of the research and propose possible directions for future work.

2. Related works

The literature on wind energy forecasting shows an evolution from traditional approaches to more advanced and combined techniques. The first approaches were based on physical methods, supported by numerical weather prediction models that simulate atmospheric dynamics. Subsequently, classical statistical time series methods (e.g., ARIMA, autoregressive models) gained ground to capture historical wind patterns. In recent years, artificial intelligence methods have emerged, including machine learning (decision trees, SVM, etc.) and deep learning (neural networks). Hanifi et al. (2020) present a critical review of these approaches, classifying them into physical, statistical (both linear models and artificial neural networks) and hybrid categories, and analyzing the factors that affect the accuracy and computational cost of each method [10]. This overview reveals that no methodology is universally superior; rather, each has strengths and weaknesses depending on the time horizon and the complexity of the data considered.

Various studies have emphasized the importance of data preprocessing to improve prediction quality. Liu and Chen [11] offer an exhaustive review of seven data processing strategies applied to wind forecasting, which include decomposing the time series into components of different frequencies, selecting and extracting relevant features, eliminating noise and outliers, and incorporating exogenous variables. These techniques aim to address the high variability and non-stationarity inherent in the wind. For example, decomposition using wavelet transforms or EMD allows the separation of long-term trends from high-frequency fluctuations, enabling models to focus on more stable patterns [11]. Similarly, variable selection through correlation analysis or specialized algorithms eliminates redundancies, while the integration of exogenous predictors (such as outputs from physical meteorological models) enriches the available information for statistical or learning models [10,11]. Beyond filter/wrapper methods, hybrid metaheuristics have shown strong results for feature selection; for example, a Sine-Cosine + Dipper-Throated hybrid reported robust performance across diverse datasets with statistical tests confirming gains [12]. Overall, preprocessing has become a crucial step in increasing the accuracy of wind energy forecasting models.

Regarding modeling methodologies, the recent trend combines machine learning and deep learning approaches with traditional techniques, giving rise to hybrid models. A notable example is that of Mohapatra et al. [13], who propose a hybrid model that integrates an ARIMA with a Kalman filter and an LSTM neural network. This combination seeks to take advantage of ARIMA's ability to capture linear and seasonal patterns, the strength of the Kalman filter in smoothing noise in the signal, and the power of LSTM to capture non-linear relationships and long-term dependencies [13]. Similarly, Du [14] introduces a novel hybrid system for short-term prediction, integrating multiple complementary algorithms—including decomposition and learning techniques—to improve the stability and accuracy of predictions [14]. On the other hand, gradient boosting algorithms have shown competitive results when adapted to time series: for instance, the application of XGBoost to wind forecasting easily incorporates exogenous variables and captures complex non-linear relationships, provided that the data is properly restructured and rigorous temporal validation is used [15]. The versatility of neural network architectures for time series prediction extends across multiple domains, demonstrating their broad applicability beyond wind energy forecasting. As an illustrative example, [16] employs a model based on Soft GRU recurrent neural networks to predict traffic congestion in smart cities, utilizing deep learning techniques to improve prediction accuracy, optimize traffic flow, and support urban management systems, achieving superior performance compared to traditional and previous deep learning approaches. Within the spectrum of deep-learning architectures, attention-enhanced generative models have emerged as a particularly effective approach for wind forecasting applications. A notable advancement in this domain is presented by Harrou et al., who developed a self-attentive variational autoencoder (SA-VAE) specifically designed for short-term wind power prediction. Their comprehensive evaluation on real turbine datasets demonstrated substantial and consistent performance improvements compared to conventional recurrent neural network architectures (RNN/LSTM/GRU) as well as standard VAE implementations, highlighting the superior capability of attention mechanisms in capturing complex temporal dependencies inherent in wind data [17]. In general, these artificial intelligence-based approaches have outperformed purely statistical methods in many cases, although their performance depends on careful calibration and the quality of the applied preprocessing.

However, most existing studies suffer from significant limitations that have been critically analyzed. A common problem is overfitting: complex learning models (especially deep networks) can overfit the training data, achieving minimal errors in historical tests but losing their generalization capability [10]. This is linked to the dependence on hyperparameters: small changes in the model configuration (number of neurons, tree depth, learning rates, etc.) can significantly alter the outcome, and many studies do not exhaustively explore this hyperparameter space, raising doubts about the robustness of their conclusions. Moreover, issues of replicability and comparability arise: different authors employ different datasets and prediction horizons, often without publishing code or fully describing their validation procedures, making independent reproduction of the results difficult [11]. A critical aspect

rarely examined is the temporal sampling of the data. The frequency at which wind data is recorded and how the training/test sets are divided can significantly influence the evaluation of the models; however, most works treat this aspect marginally or assume it to be fixed. Effenberger et al. [9] highlight this issue by systematically examining how temporal resolution affects long-term predictions: their study shows that using data aggregated in overly broad intervals (e.g., daily or monthly averages) distorts the actual distribution of wind speeds and exacerbates power prediction errors, whereas more frequent sampling (for example, instantaneous values every 3–6 hours) adequately preserves the variability necessary for reliable forecasts [9]. This finding underscores the importance of considering sampling as an integral part of experimental design, something that most previous research has not addressed with sufficient rigor.

3. Methodology and resources

3.1. Resources and data description

The dataset used in this study comes from the Kelmarsh wind farm, located in the UK, and is publicly available through Zenodo [18]. It includes data from six Senvion MM92 turbines, each with a rated power of 2.05 MW, a rotor diameter of 92.5 meters, and a hub height of 100 meters. The dataset comprises high-resolution SCADA (Supervisory Control and Data Acquisition) information recorded every 10 minutes, collected from 2016 to mid-2021. It contains key operational parameters such as wind speed, generated power, turbine status indicators, ambient temperature, and rotor speeds. In addition, it includes static information such as turbine coordinates, technical specifications, site substation data, and, where available, data from the fiscal meter.

During the pre-processing stage, the variables in the dataset were categorized into two groups: turbine-specific variables and external environmental variables. Turbine-specific variables, such as rotor speeds, nacelle positions, and turbine status indicators, reflect the internal operational state of each turbine and are essential for performance monitoring and maintenance diagnostics. These variables focus primarily on the internal conditions of the turbine rather than on external factors that influence power generation. However, external variables such as wind speed, wind direction, and ambient temperature directly affect the amount of power generated, as they determine the available kinetic energy for conversion [4]. In this analysis, wind speed, wind direction, and ambient temperature were selected as key exogenous variables to predict generated power, given their significant impact on variations in wind energy generation.

3.2. Machine learning approaches

The choice of a prediction model fundamentally depends on the nature of the data and the specific objectives of the analysis. In this case, the data pertain to time series related to wind energy prediction, which implies working with time-structured information and significant temporal dependencies. In the following, the theoretical foundation of the selected models and their applicability to time series analysis is presented, along with the reasons that justify their selection.

SARIMAX (Seasonal AutoRegressive Integrated Moving Average with eXogenous regressors)

The SARIMAX model combines several components to enhance its predictive capabilities. The autoregressive (AR) component captures the influence of past values on current observation, allowing the model to identify patterns where the present output depends on its historical values. The integration (I) component is employed to eliminate non-stationarity by transforming the series into one where statistical properties such as mean and variance remain constant over time. Moving average (MA) terms focus on modeling past prediction errors and smoothing out random fluctuations to provide more stable forecasts [19].

To handle seasonality, SARIMAX incorporates periodic adjustments through seasonal AR, differentiation and MA terms, denoted as P, D, and Q, respectively, which allow the model to capture repetitive patterns occurring within a defined seasonal cycle of length s. An important enhancement in SARIMAX is the inclusion of exogenous variables (X_t), enabling the model to integrate external data sources that influence the target variable. These exogenous inputs can include time-dependent factors that provide additional context, such as meteorological data for wind energy forecasting, macroeconomic indicators for financial models, or other domain-specific factors. The SARIMAX model is defined by the following equation:

$$\Phi_P(L^s)\phi_p(L)(1-L^s)^D(1-L)^d Y_t = \Theta_Q(L^s)\theta_q(L)\epsilon_t + \beta X_t \tag{1}$$

Here:

- $\phi_p(L)$ and $\theta_q(L)$: Non-seasonal AR and MA polynomials, respectively.
- $\Phi_P(L^s)$ y $\Theta_Q(L^s)$: Seasonal AR and MA polynomials with delay s.
- $(1-L)^d$ y $(1-L^s)^D$: Differencing operators applied to achieve stationarity in the non-seasonal and seasonal components.
- βX_t: Represents the effect of the exogenous variables.
- ϵ_t: The error term that captures the residuals unexplained by the model.

To effectively implement a SARIMAX model, it is essential to optimize the following parameters:

- Non-seasonal components: p (autoregressive), d (differencing), q (moving average).
- Seasonal components: P (seasonal autoregressive), D (seasonal differencing), Q (seasonal moving average), and s (length of the seasonal period).
- Exogenous variables (X_t): These should be carefully selected based on their relevance and influence on the target time series.

SARIMAX is employed for its ability to simultaneously handle seasonality, non-stationarity, and the influence of external factors within a time series. This makes it particularly useful in contexts where the series exhibit regular cycles and are affected by external variables, as is the case with wind energy. Its flexibility allows the capture of complex patterns in the data, providing a robust and effective model for predictive analysis.

XGBoost (eXtreme Gradient Boosting)

XGBoost is a supervised learning algorithm based on the gradient boosting method, an ensemble technique that builds multiple models, typically decision trees, in a sequential manner [20]. In each iteration, the new model is trained to correct the errors made by the previous models, thereby progressively improving the prediction accuracy. Its design is optimized for speed and performance, making it a highly efficient tool for handling large datasets and high-dimensional problems. The model makes predictions by adding functions from a set of decision trees. The general equation is:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), \quad f_k \in \mathcal{F} \tag{2}$$

Where:

- $\hat{y}_i$: Prediction for instance i,
- K: Total number of trees,
- $f_k(x_i)$: Prediction of the k-th tree for instance x_i,
- $\mathcal{F}$: The space of tree functions, defined as:

$$\mathcal{F} = \{f(x) = w_{q(x)}\}$$

Here:
- $q(x)$: The tree structure that assigns a leaf node to each instance x,
- w: The values at the leaves.

The objective of XGBoost is to minimize the following loss function:

$$\mathcal{L} = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k), \tag{3}$$

Where:

- $l(y_i, \hat{y}_i)$ is the loss function (for example, the mean squared error),
- $\Omega(f_k)$ is the regularization term used to control the complexity of the model, defined as:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$$

Here:
- γ: Penalizes the number of leaves (T),
- λ: Regulates leaf weights w_j.

XGBoost is used for various reasons, including its optimized design that enables rapid training even on large datasets, and its computational efficiency. Moreover, by using gradient boosting, the algorithm is able to capture complex patterns in the data, and the incorporation of regularization terms into its objective function helps prevent overfitting. Although XGBoost is not specifically designed for time series, it can indirectly model temporal patterns by incorporating engineered features such as lags or trends [15].

Hybrid model

A hybrid model has been proposed that combines the strengths of different techniques to address the inherent complexities of time series forecasting, specifically for the generation of wind energy. This model includes three main components:

1. **EMD Filter (Empirical Mode Decomposition):**
This is a decomposition technique used to break down the data into multiple oscillatory components known as Intrinsic Mode Functions (IMFs). These IMFs represent the different frequencies present in the series, separating high-frequency patterns (noise) from low-frequency ones (trends). The decomposition simplifies the analysis by reducing the complexity of the time series, allowing subsequent models to focus on clearer and more specific patterns.
The EMD decomposition process is based on the general formula:

$$X(t) = \sum_{i=1}^{n} \mathrm{IMF}_i(t) + r_n(t) \tag{4}$$

Where:
- $X(t)$ is the original time series.
- $\mathrm{IMF}_i(t)$ is the i-th Intrinsic Mode Function, representing an oscillatory component of the series.
- $r_n(t)$ is the final residue that captures the overall non-oscillatory trend after all IMFs have been extracted.
- n is the total number of IMFs generated during the process.

The EMD method employs an iterative process to identify each $\mathrm{IMF}_i(t)$, progressively separating the components of different frequencies in the time series.
EMD is known to exhibit boundary effects that may distort the first and last samples of each decomposed segment. In this implementation, explicit boundary padding or mirror extension was not applied. Two design choices mitigate the practical impact on the results: (i) EMD is applied only on the training split to derive IMFs that are used to learn latent sequential features with the LSTM; validation/test predictions are produced downstream by XGBoost from the learned representations and exogenous inputs, rather than by directly using IMF endpoints from the evaluated segments; and (ii) features are computed on 50% -overlapping windows and mapped back to the timeline via overlap-average (each sample aggregates contributions from all overlapping windows and is normalized by the overlap count), which down-weights edge artifacts.
2. **LSTM (Long Short-Term Memory) with Attention Layer:**
LSTMs are a variant of Recurrent Neural Networks (RNNs), designed specifically for processing sequential or temporal data. They use recurrent connections to maintain a "memory state" that enables them to model temporal dependencies, such as those found in time series or text. Traditional RNNs often struggle to capture long-term dependencies due to issues like vanishing or exploding gradients. LSTMs overcome this by incorporating memory cells and gating mechanisms that control the flow of information, allowing them to retain relevant data while discarding irrelevant information [21].
The key equations of an LSTM are:
- **Forget Gate (f_t):** Determines which information from the previous memory cell should be discarded.

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{5}$$

- **Input Gate (i_t):** Determines what new information should be stored in the memory cell.

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{6}$$

- **Candidate State ($\tilde{C}_t$):** Generates the new candidate content for the memory cell.

$$\tilde{C}_t = \tanh(W_C x_t + U_C h_{t-1} + b_C) \tag{7}$$

- **Memory Cell State (C_t):** Updates the memory cell by combining the previous state, the candidate state, and the corresponding gates.

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{8}$$

- **Output Gate (o_t):** Determines which information from the memory cell should be used in the output.

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{9}$$

- **Hidden State (h_t):** Generates the output of the LSTM by combining the output gate and the memory cell state.

$$h_t = o_t \odot \tanh(C_t) \tag{10}$$

Where:
- x_t is the input at time t,
- h_{t-1} is the previous hidden state,
- C_{t-1} is the previous memory cell state,
- f_t, i_t, o_t are the forget, input, and output gates respectively,
- C_t is the updated memory cell state,
- W_f, W_i, W_C, W_o are the weights associated with the input,
- U_f, U_i, U_C, U_o are the weights associated with the previous hidden state,
- b_f, b_i, b_C, b_o are the biases,
- σ is the activation function,
- tanh is the hyperbolic tangent function,
- $\odot$ denotes element-wise multiplication.

By integrating an attention layer, the LSTM's ability to identify and prioritize the most relevant relationships in sequential data is enhanced. The attention layer calculates a weight (α_t) for each time step, indicating the relative importance of each input. This is achieved through:

- **Calculation of Attention Weights (α_t):**

$$\alpha_t = \frac{\exp(e_t)}{\sum_{k=1}^{T} \exp(e_k)} \tag{11}$$

Where e_t is the relevance score calculated as:

$$e_t = v^\top \tanh(W_a h_t + b_a) \tag{12}$$

- **Calculation of the Context (c_t):** Combines the inputs weighted by the attention weights.

$$c_t = \sum_{t=1}^{T} \alpha_t h_t \tag{13}$$

Where:
- T is the sequence length,
- h_t is the LSTM output at time t,
- W_a, b_a, v are the parameters of the attention layer,
- α_t is the attention weight for time step t,
- c_t is the context vector generated by the attention layer.

3. **XGBoost (eXtreme Gradient Boosting):**
Once the temporal features have been extracted by the LSTM, XGBoost is used to model the relationships between these features and the target values. Its ability to handle high-dimensional data and its computational efficiency make it ideal for combining with deep learning models.

The described model can be considered a pipeline that integrates multiple stages of data transformation and modeling to optimize predictions. It combines the strengths of various mathematical methods, such as frequency decomposition, deep learning (DL), and decision trees. This design enables the capture of both temporal patterns and complex relationships between features without excessively increasing computational costs. The application of the EMD filter separates the frequencies in the time series, reducing noise and improving the quality of the predictions. Meanwhile, the attention layer in the LSTM networks enhances the model by identifying and prioritizing the most relevant parts of the time series. The LSTM outputs are processed through an overlap-add technique to generate temporal features $\hat{I}_1(t), \ldots, \hat{I}_m(t)$ for each timestamp t. These extracted features are then concatenated with the raw exogenous variables $x_{\text{exog}}(t)$ to construct the input feature matrix for XGBoost. This concatenation approach provides a direct fusion strategy without requiring additional learned parameters. For each timestamp t, the XGBoost input row is

$$\mathbf{X}_t = \left[\hat{I}_1(t), \ldots, \hat{I}_m(t), x_{\text{exog}}(t)\right], \tag{14}$$

where $\hat{I}_1(t), \ldots, \hat{I}_m(t)$ represent the temporal features extracted from the LSTM networks and $x_{\text{exog}}(t)$ denotes the exogenous variables at time t. XGBoost then complements the pipeline by efficiently modeling nonlinear relationships in the data [22] [23].

To isolate sampling-rate effects, the standalone XGBoost baseline employs compact exogenous inputs with consistent imputation across all models. Temporal structure is captured in the hybrid architecture through the upstream EMD-LSTM module, which generates temporal features for XGBoost. This design choice enables fair comparison of sampling granularity impacts while acknowledging that enhanced feature engineering (lagged variables, rolling statistics, calendar features) and native missing value handling could further improve XGBoost performance.

The study aimed to isolate the effects of data granularity and exogenous-variable choice; to avoid confounding from a large hyperparameter sweep, a stability-oriented training recipe and a conservative boosting setup were adopted, with validation MAE monitored throughout. An XGBoost pre-calibration phase performed a time-series cross-validated grid search on a resampled training set (scored by MAE) to identify a stable configuration, which was then fixed across all hybrid experiments. During boosting, validation MAE was tracked across rounds to guard against overfitting, and a shallow, regularized configuration with learning-rate shrinkage and subsampling was used to maintain stability while keeping compute within scope. For the LSTM component, a validation-guided procedure-early stopping, learning-rate reduction on plateau, and best-model checkpointing-was employed together with a fixed windowing scheme (constant overlap) and standard regularization layers, limiting sensitivity without an exhaustive sweep. In addition, a reconstruction objective on the EMD components was coupled with the same validation controls, with the overall design intended to dampen sensitivity to step size, training length, and capacity without broad hyperparameter exploration.

The proposed hybrid architecture offers significant interpretability advantages through its hierarchical analytical structure. The EMD decomposition facilitates frequency-domain analysis by decomposing the input signal into distinct temporal components, enabling identification of the relative contributions of different time scales to forecast performance. The attention mechanism within the LSTM network provides temporal attribution maps that quantify the importance of historical information for current predictions, thereby revealing the temporal dependencies captured by the model. The XGBoost component enables detailed feature attribution through SHAP (SHapley Additive exPlanations) analysis, which decomposes individual predictions into additive contributions from each input variable. Unlike global feature importance measures, SHAP provides instance-specific explanations by calculating the marginal contribution of each feature relative to the model's expected output [24]. This approach allows for the decomposition of predictions into constituent parts, attributing specific numerical values to both exogenous variables and learned temporal features. The resulting interpretability framework supports model validation, facilitates understanding of prediction mechanisms, and enhances confidence in model outputs for practical applications.

4. Data preprocessing and exploratory analysis

4.1. Data cleaning and transformation

Although the dataset documentation [18] did not explicitly mention temporal lags, the detected gaps in the signals indicated possible irregularities. To address this issue, a two-stage strategy was implemented: linear interpolation and hourly mean imputation. Linear interpolation was used to estimate missing values using adjacent data [25], while for the remaining gaps, the data were grouped by hour of the day, filling missing values with the corresponding hourly mean. Linear interpolation was chosen for short gaps as it preserves local shape without introducing artificial oscillations that spline methods might generate. For longer gaps, hour-of-day mean imputation maintains diurnal cycles while avoiding synthetic high-frequency content that could mislead temporal models. This ensured the preservation of natural diurnal patterns in wind speed and turbine operations, maintaining consistency with expected temporal trends. Although the reconstructed values do not exactly replicate the original time series, this limitation is unlikely to significantly impact model performance. This preprocessing strategy was designed to prioritize the capture of general trends and key relationships in the data, rather than focusing on specific patterns, thus reducing the risk of overfitting by avoiding the memorization of irrelevant details. As a result, reliable predictions are achieved even in scenarios with temporal inconsistencies.

Wind speed, as a time-dependent variable, exhibits highly unstable behavior due to natural atmospheric dynamics. Traditional methods of handling outliers might remove significant fluctuations that are essential to understand the true wind conditions affecting energy generation. Eliminating these extreme values could result in the loss of valuable information, as such variations are an intrinsic feature of wind patterns. For this reason, it was decided to preserve these fluctuations rather than apply conventional outlier removal techniques. This approach is

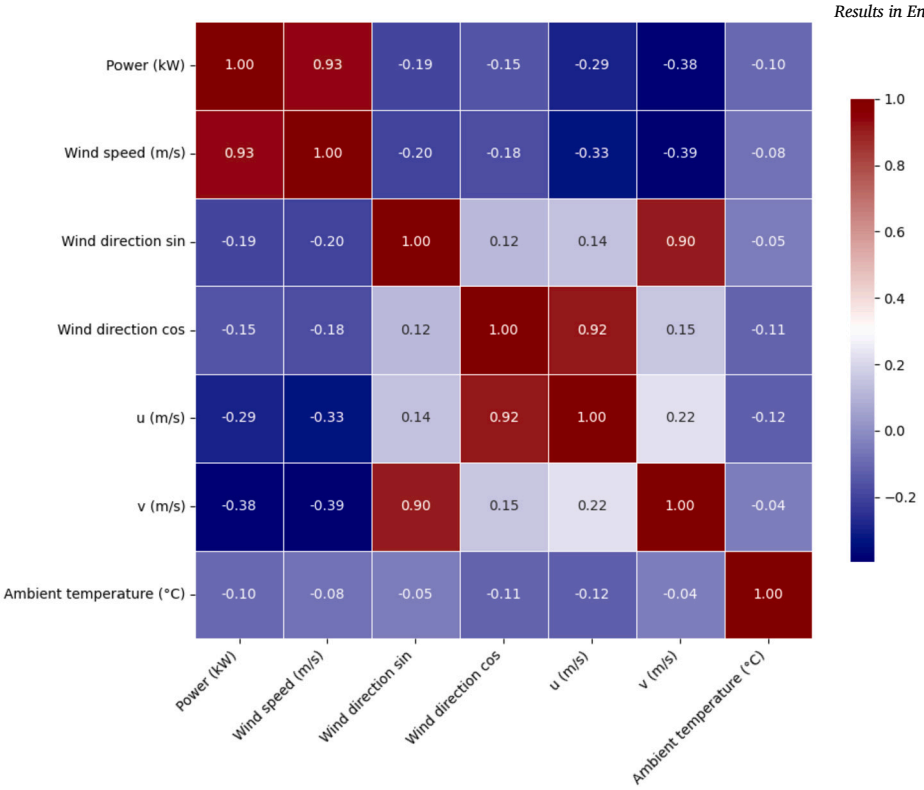


Fig. 1. Correlation matrix of the predictor variables, represented as a heatmap.

supported by the review by Zou et al. [26], which emphasizes the importance of maintaining natural variability in wind speed data to ensure accurate analysis and modeling.

However, the wind direction variable was transformed using its sine and cosine representations, equivalent to a projection onto the Cartesian plane. This transformation decomposes the wind direction into its u (x-axis) and v (y-axis) components, facilitating analysis in terms of vectors. This treatment allows working with circular directions by eliminating issues associated with the cyclic nature of angles (for example, the jump from 359^o to 0^o). By expressing the wind direction in these components, a continuous representation is obtained that coherently integrates both the direction and the magnitude of the wind, thereby enhancing its modeling and analysis.

This study focused on predicting wind energy at the wind farm level, prioritizing overall energy production rather than individual analysis of each turbine. To achieve this, data from the six turbines were integrated, consolidating the metrics into a single dataset that represents the global performance of the wind farm. The total power and energy production was determined by adding the corresponding values of all the turbines. For external variables such as wind speed, wind direction, and ambient temperature, averages were calculated across the turbines to obtain a dataset representative of the entire farm.

Finally, the dataset was normalized to ensure that variables with different scales did not disproportionately influence the forecasting model. Standard normalization (z-score), defined by equation (15), adjusts the data to a distribution with a mean of zero and a standard deviation of one, ensuring that all variables have the same weight in the model:

$$X_{normalized} = \frac{X - \mu}{\sigma} \tag{15}$$

This procedure transforms each variable by subtracting its mean and dividing by its standard deviation, centering it around zero with unit variance. Without this step, variables with larger numeric ranges could dominate the model's behavior, introducing biases and negatively affecting prediction accuracy. Normalization guarantees a balanced representation of all features, thereby improving the model's stability and performance.

Table 1
Mutual information measures between the predictor variables and energy production.

Variable	Mutual Information
Wind Speed (m/s)	1.96
v (m/s)	0.55
u (m/s)	0.51
Ambient Temperature (^{o}C)	0.11
Wind Direction (sine)	0.10
Wind Direction (cosine)	0.10

4.2. Exploratory data analysis

A preliminary analysis was conducted to evaluate the relevance of external variables using correlation metrics and mutual information. As shown in Fig. 1, the correlation matrix of all variables in the dataset has been represented with a heat map. This means that the linear relationships between pairs of variables are encoded on a color scale, with more intense hues representing stronger correlations, whether positive or negative. The most significant values are those corresponding to power and wind speed. Since power is a direct representation of energy, wind speed is identified as the most influential variable. This finding is corroborated by Table 1, where the mutual information results align with those observed in the heatmap. However, it is still possible that the Cartesian components v and u provide valuable information to the model, although less direct than wind speed.

The results of the analysis highlight that wind speed is the main driver of energy generation, exhibiting the strongest correlation with generated power. In contrast, other external factors, such as ambient temperature and wind direction, show much weaker associations, suggesting that their contribution to the model's predictive capability is lower. Based on these findings, the analysis focused on the wind speed

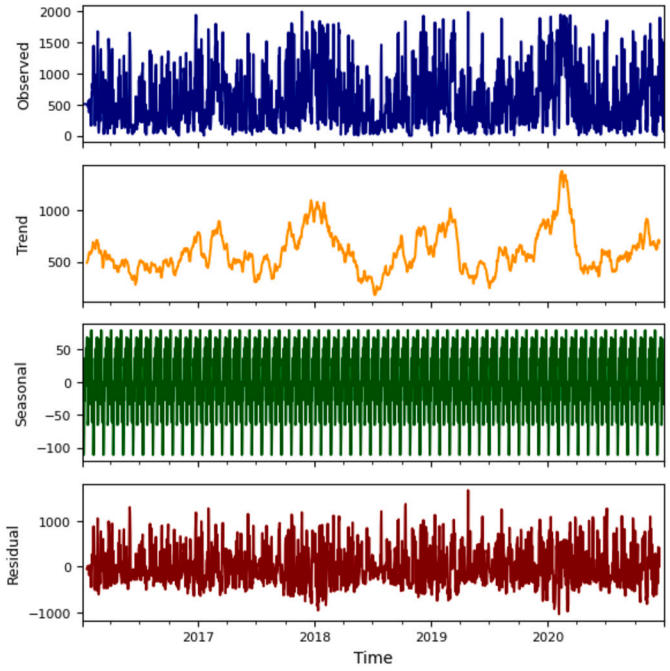


Fig. 2. Seasonal decomposition of wind energy generation.

as the key exogenous variable, along with its relationship to the generated power, to optimize the model.

To further examine the series, another important analysis was conducted to assess both its stationarity and seasonality. Stationarity is defined as the property of a time series whose statistical characteristics—such as mean, variance, and autocorrelation—remain constant over time. In contrast, seasonality refers to the presence of repetitive, predictable patterns that occur at regular intervals, such as seasonal weather changes or hourly variations in energy demand. If a series were strictly stationary, it could not exhibit seasonality, since seasonal effects introduce cyclical and predictable changes in the mean or variance, thereby violating the key condition of statistical invariance over time.

To analyze the stationarity and seasonality of the series, a seasonal decomposition was carried out, with its results presented in Fig. 2. This process allows the time-series data to be decomposed into its main components: trend, seasonality, and residuals. Using an additive model, which assumes that these components sum up to form the observed data, it is possible to isolate the recurring seasonal patterns that occur over a specified period, as well as to identify underlying trends and irregular fluctuations [27].

For this type of analysis, the Augmented Dickey-Fuller (ADF) test and the Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test are commonly employed [28]. These tests are usually applied together to obtain a more complete diagnosis regarding the series' stationarity.

The ADF test evaluates the null hypothesis that a series has a unit root (i.e., it is non-stationary) against the alternative hypothesis that the series is stationary, as defined in Equation (16). A low p-value leads to rejecting the null hypothesis, suggesting stationarity.

$$\Delta y_t = \alpha + \beta t + \gamma y_{t-1} + \sum_{i=1}^{p} \delta_i \Delta y_{t-i} + \epsilon_t \tag{16}$$

Here, Δy_t is the difference of y_t, α is a constant, βt represents a trend term, γ indicates the presence of a unit root, $\sum_{i=1}^{p} \delta_i \Delta y_{t-i}$ controls for autocorrelation, and ϵ_t is the error term.

On the other hand, the KPSS test evaluates the null hypothesis that a series is stationary around a deterministic trend against the alternative hypothesis of non-stationarity, as described in Equation (17). This test calculates a statistic based on the cumulative sum of the residuals

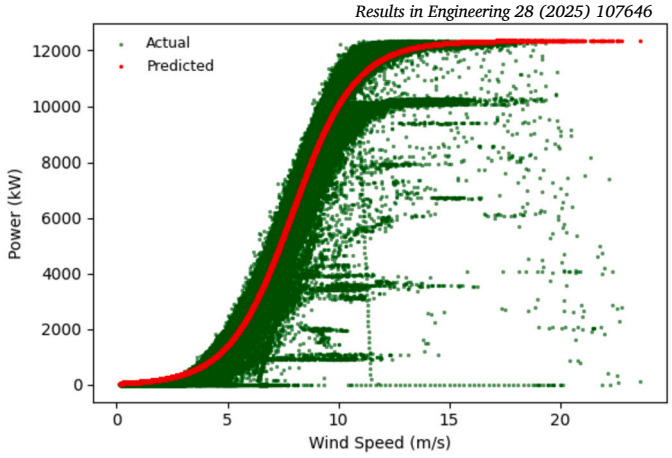


Fig. 3. Sigmoidal curve fitted to represent the relationship between wind speed and generated power.

from a regression. A low p-value in this test leads to rejecting the null hypothesis, which suggests non-stationarity.

$$y_t = r_t + \beta t + \epsilon_t \tag{17}$$

In this equation, r_t is a stochastic trend, βt is a deterministic trend, and ϵ_t is the error term.

In our analysis, after decomposing the series seasonally (see Fig. 2), the ADF test suggested that the time series is stationary, while the KPSS test indicated non-stationarity. This discrepancy may be explained by the influence of seasonal patterns: although the overall variation in the seasonal component is slight and recurring, it might still be sufficient to affect the stationarity tests differently. Thus, the series might exhibit characteristics of both stationarity and non-stationarity, highlighting the complex interplay between its underlying trends and seasonal patterns.

The analysis of the relationship between wind speed and generated power revealed a sigmoidal curve pattern, which is characteristic of wind turbine performance. At low wind speeds, the force is insufficient to efficiently turn the blades, so the generated power remains very low. However, once the wind speed exceeds a certain threshold—known as the cut-in speed—the turbine begins to generate power more rapidly. This growth continues until the rated speed is reached, at which point the turbine attains its maximum power. Beyond this point, power generation stabilizes, as turbines are designed to limit their output to prevent mechanical wear and ensure safe operation even if wind speeds continue to increase [29].

The sigmoidal function is particularly well-suited to model this relationship, as it captures the initial slow increase, the rapid growth in power generation, and the subsequent saturation. This curve can be fitted using the logistic function, mathematically expressed as:

$$P(w) = \frac{P_{max}}{1 + e^{-k(w - w_0)}} \tag{18}$$

where $P(w)$ is the predicted power for a wind speed w, P_{max} is the maximum generated power, k controls the steepness of the curve, and w_0 represents the wind speed at the inflection point where the power generation increases most rapidly.

Fig. 3 illustrates how the logistic function models the relationship between wind speed and the power generated by the turbine. Fitting the curve to the data allows for an accurate representation of the turbine's performance characteristics, facilitating the integration of wind speed data into forecasting models. To fit the logistic function, a nonlinear least squares method was used. This technique minimizes the sum of the squared residuals, where each residual represents the difference between the observed power and the value predicted by the function. The optimization process iteratively adjusts the parameters P_{max}, k and w_0 to find the best-fitting curve that minimizes the error.

5. Application and performance evaluation

5.1. Baseline

To evaluate the performance of the models, three key time horizons were considered: short-term (3 days), medium-term (1 week), and long-term (1 month). These horizons allow us to analyze how the model behaves at different temporal scales, from immediate forecasts to more extended projections. Additionally, three different data sampling frequencies were established—10 minutes, 30 minutes, and 60 minutes—to assess the impact of data granularity on model performance, considering both more detailed predictions and less frequent approximations. Furthermore, as part of the analysis, the impact of including exogenous variables was evaluated. In particular, a comparison was made between using the power curve and wind speed as input to the model, with the objective of determining whether this exogenous variable can offer an improvement in predictive capability. The metrics used to evaluate model performance are described in detail below.

- **Coefficient of Determination (R^2):**
  This metric measures how well the model explains the variation in the data. An R^2 value close to 1 indicates that the model explains most of the variability, while a value near 0 indicates the opposite. It is given by:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2} \tag{19}$$

  where y_i are the actual values, $\hat{y}_i$ are the predicted values, $\overline{y}$ is the mean of the actual values and n is the number of observations.
- **Mean Absolute Error (MAE):**
  This represents the average of the absolute errors between the actual and predicted values. It measures the average magnitude of the errors without considering their direction. It is defined as:

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{20}$$

- **Root Mean Squared Error (RMSE):**
  This metric calculates the square root of the mean of the squared errors, giving more weight to larger errors due to its quadratic nature. It is especially useful for identifying cases where large deviations are problematic:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{21}$$

- **Mean Absolute Percentage Error (MAPE):**
  This measures the absolute error in percentage terms, which allows for the interpretation of error relative to the size of the actual values. It is calculated as:

$$\mathrm{MAPE} = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{22}$$

- **Symmetric Mean Absolute Percentage Error (sMAPE):**
  This is a modified version of MAPE, designed to correct asymmetry issues in percentage errors. It is particularly useful when both actual and predicted values are close to zero, as it avoids divisions by small numbers. Its formula is:

$$\mathrm{sMAPE} = \frac{100}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{(|y_i| + |\hat{y}_i|)/2} \tag{23}$$

As a starting point, a baseline model was established under the assumption that the behavior in the upcoming year would be identical to that of the previous year—in other words, predictions were made by assuming that last year's values match those of the current year. The results of this baseline model are presented in Table 2, serving as a reference for evaluating the performance of the developed models.

Table 2
Baseline model results for different time horizons.

Time Horizon	R^2	MAE	RMSE	MAPE	sMAPE
3 days	1.54	1.43	1.75	3.67	2.05
1 week	0.65	1.13	1.46	4.32	0.64
1 month	0.47	1.12	1.44	4.86	1.08

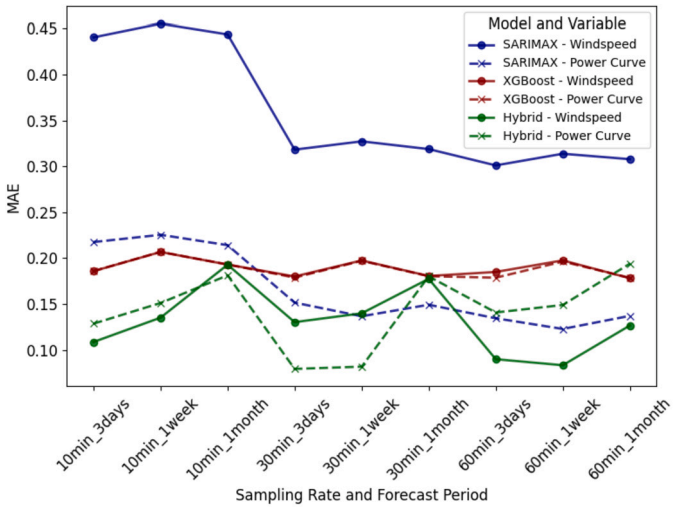


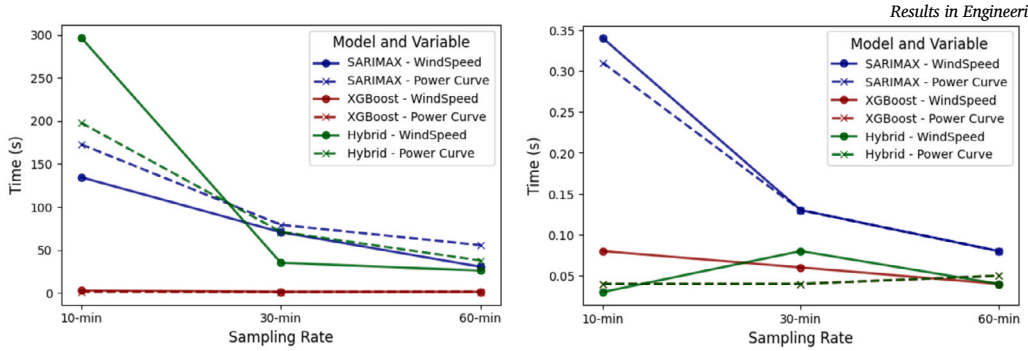
Fig. 4. Comparison of Mean Absolute Error (MAE) by model and exogenous variable, broken down by sampling rate and time horizons.

5.2. Exogenous variables

In Fig. 4 the MAE for each model and exogenous variable is displayed, broken down by sampling rate and time horizon. This figure provides a clear and comprehensive view of the effect of the exogenous variables feeding the model, as well as the impact of data sampling rates. A lower MAE reflects better model performance, indicating a lower average error in the predictions.

It is observed that the SARIMAX model experiences a significant improvement when incorporating the power curve as an exogenous variable compared to using only wind speed. This indicates that the power curve more directly and representatively reflects the relationships with the target variable, favoring a better model fit. On the other hand, the XGBoost model shows relatively consistent results under both configurations. SARIMAX, based on linear relationships and statistical assumptions, benefits particularly from a variable that simplifies and synthesizes the underlying dynamics, as the power curve does [30]. In contrast, the XGBoost model, being based on decision trees, exhibits greater flexibility in directly modeling complex relationships, allowing it to maintain robust performance regardless of the exogenous variable used.

In the case of the hybrid model, a higher sensitivity to both the sampling rate and the prediction time horizon is observed. Specifically, wind speed stands out for short-term horizons (3 days) and broader sampling rates (60 minutes), achieving lower values of MAE, MAPE, and sMAPE. Meanwhile, the power curve provides competitive performance with intermediate sampling rates (30 minutes) and proves particularly effective for short- and medium-term horizons. These results can be explained by the structure and operation of the hybrid model. Its effectiveness largely depends on the nature of the exogenous variable used. In the case of wind speed, the EMD filter and LSTM effectively capture its rapid temporal variations, which favors short-term horizons and broader sampling rates. Conversely, the power curve, being a more direct and smoothed



(a) Training Time. (b) Prediction Time.

Fig. 5. Comparison of training (5a) and prediction (5b) times for the models under different sampling rate configurations.

representation of the generated power, allows the model to better leverage intermediate sampling rates and achieve more precise predictions in short- and medium-term horizons.

The observed preference for different exogenous inputs across models can be explained by their fundamental assumptions and architectures. SARIMAX benefits from the power curve because exogenous inputs enter linearly in this model, and the wind-power relationship is strongly nonlinear (sigmoidal). Providing the power-curve transform effectively linearizes this relationship for SARIMAX, reducing misspecification and improving fit. In contrast, XGBoost and the hybrid model prefer raw wind speed because these nonlinear learners can model nonlinearities and interactions directly. Using raw wind speed preserves high- and mid-frequency variability (ramps, turbulence, threshold effects) that a fitted power curve smooths out, and these fluctuations are informative for tree splits and for the LSTM's temporal features. Additionally, the hybrid model learns on IMF sequences from EMD, where raw wind speed yields richer multi-scale IMFs compared to the smoother power-curve proxy.

5.3. Impact of data granularity

The impact of the sampling rate is particularly noticeable in terms of computational efficiency. However, in this section the focus has been on analyzing the direct performance of the models—that is, the obtained metrics. Within the sub-hourly-to-hourly regime available in the SCADA data, data granularity materially affects model performance. As observed in the Fig. 4, the SARIMAX model exhibits a considerable improvement in performance as the sampling rate increases (i.e., with wider intervals). This can be explained by the fact that a lower level of data granularity smooths out variations, reducing noise in the time series, which is particularly beneficial for statistical models such as SARIMAX, which are more sensitive to such disturbances.

On the other hand, the XGBoost model demonstrates more consistent performance across different sampling rate configurations, although a slight deterioration is evident when using wider intervals (60 minutes), especially in long-term predictions. This behavior reflects XGBoost's ability to handle more complex and noisy data, although it also suggests that the model benefits from higher granularity (more frequent rates) to optimize the precision of its predictions.

Finally, the hybrid model shows a pattern similar to that of XGBoost, exhibiting balanced performance with moderate sampling rates (30 minutes). However, unlike SARIMAX, this model is less susceptible to noise, thanks to the integration of both statistical and machine learning approaches. Its architecture—which combines EMD decomposition, an LSTM with an attention layer, and the XGBoost model—allows it to better adapt to the specific characteristics of the data. This combination balances smoothness and granularity, maximizing prediction accuracy. This behavior highlights the hybrid model's ability to integrate the strengths of both methods and efficiently respond to different sampling rate configurations.

5.4. Computational efficiency evaluation

In Fig. 5 the training and prediction times are presented. As observed, the SARIMAX model experiences a significant decrease in training times as the sampling rate increases—indicating that lower data granularity simplifies the model fitting process. Additionally, incorporating the power curve as an exogenous variable slightly increases the training times compared to using wind speed, likely due to the added complexity of this variable. In terms of prediction, SARIMAX maintains consistently low times, reaching a maximum of 0.34 seconds, which highlights its efficiency in this task.

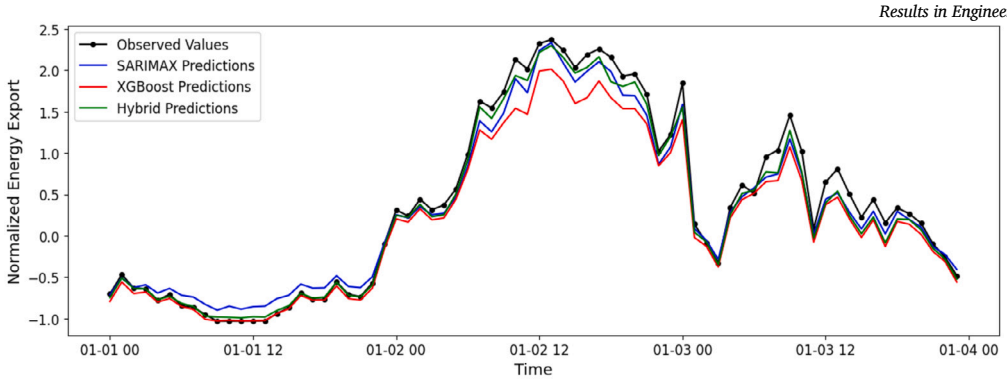
Conversely, the XGBoost model stands out as the most computationally efficient, both in training and prediction. Its training times are significantly lower compared to the other models, and the impact of the sampling rate is less pronounced. Furthermore, using the power curve as an exogenous variable further reduces its training times. Regarding prediction, XGBoost achieves extremely low times—up to a maximum of 0.08 seconds—making it an ideal choice for real-time applications.

The hybrid model, however, exhibits the highest training times, reflecting its greater complexity. Training this model with wind speed at a 10-minute sampling rate requires 296.85 seconds, but this time decreases drastically to 26.05 seconds with a 60-minute sampling rate. Similar to SARIMAX, the use of the power curve as an exogenous variable slightly increases the training times. Nonetheless, despite its high computational cost during training, the hybrid model compensates with very low prediction times, reaching a maximum of 0.08 seconds—comparable to XGBoost—making it competitive for rapid prediction tasks.

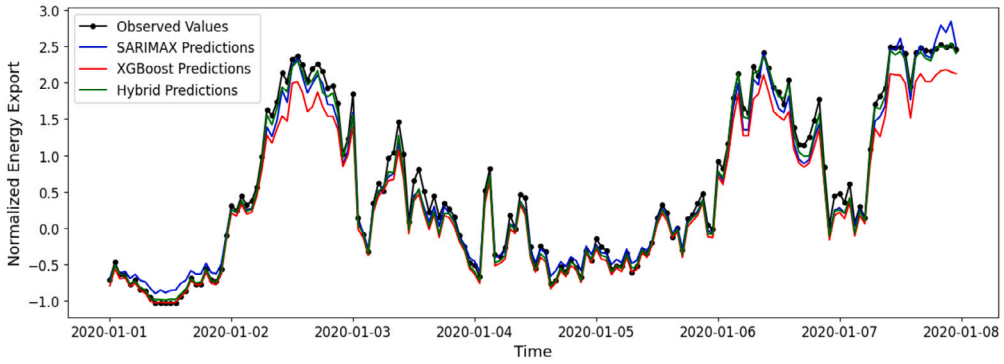
5.5. Time horizon analysis

The performance analysis over different time horizons was conducted by selecting the best configuration for each model. The optimal configuration found for the SARIMAX model employs the power curve as an exogenous variable along with a sampling rate of 60 minutes. In contrast, both the XGBoost and hybrid models achieved their best performance using wind speed as the exogenous variable, also with a sampling rate of 60 minutes. It is important to note that in the case of the XGBoost model, either of the proposed exogenous variables (power curve or wind speed) can be used interchangeably, as well as a sampling rate of either 30 or 60 minutes. This behavior reflects its consistency across different configurations, allowing for greater flexibility in its application.

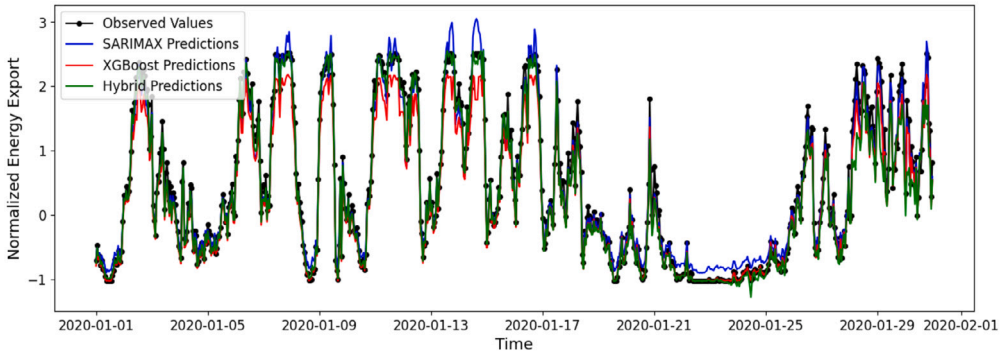
The performance of the models on the test data, broken down by the considered time horizons, is illustrated in Fig. 6. The metrics associated with the optimal configuration of each model are detailed in Table 3, providing a clear framework to analyze their performance in different



(a) Time Horizon: 3 days.



(b) Time Horizon: 1 week.



(c) Time Horizon: 1 month.

Fig. 6. Performance of the models with their optimal configurations in short-term (6a), medium-term (6b), and long-term (6c) predictions.

Table 3
Performance of the models with their optimal configurations over different time horizons.

Model	Time Horizon	R^2	MAE	RMSE	MAPE	sMAPE
SARIMAX	3 días	0.98	0.13	0.16	0.19	0.23
	1 semana	0.98	0.12	0.15	0.29	0.25
	1 mes	0.98	0.14	0.16	0.39	0.25
XGBoost	3 días	0.95	0.18	0.25	0.30	0.36
	1 semana	0.95	0.20	0.25	0.57	0.39
	1 mes	0.96	0.18	0.24	0.62	0.33
Híbrido	3 días	0.99	0.09	0.12	0.18	0.22
	1 semana	0.99	0.08	0.11	0.29	0.24
	1 mes	0.97	0.13	0.21	0.54	0.28

prediction scenarios. The SARIMAX model shows consistently stable performance across all time horizons. With a constant R^2 of 0.98, this model demonstrates a strong ability to explain the variability of the data. The absolute and relative error metrics exhibit slight variations: the MAE and RMSE remain low (0.12–0.14 and 0.15–0.16, respectively), indicating a precise fit. Although the MAPE and sMAPE increase slightly with the horizon (from 0.19 to 0.39 for MAPE and from 0.23 to 0.25 for sMAPE), this increase is moderate, ensuring that SARIMAX remains reliable even in long-term predictions.

The XGBoost model, although performing slightly inferior to SARIMAX, also shows a high R^2, between 0.95 and 0.96. However, its absolute and relative error metrics are more sensitive to the increase in the time horizon. The MAE varies between 0.18 and 0.20, and the RMSE fluctuates between 0.25 for 3 days and 0.24 for 1 month. The relative metrics, such as MAPE and sMAPE, exhibit a more pronounced deterioration with the horizon (MAPE increasing from 0.30 to 0.62 and sMAPE varying from 0.36 to 0.33), which suggests that XGBoost may lose precision in long-term predictions.

On the other hand, the hybrid model stands out for its excellent performance, especially over short- and medium-term horizons. In 3-day and 1-week predictions, it achieves an R^2 of 0.99, indicating an al-

most perfect fit. Furthermore, its absolute error metrics, such as MAE (0.09–0.13) and RMSE (0.12–0.21), are considerably lower than those of the other models, reflecting high precision. However, in the 1-month horizon, its relative performance slightly declines, with increases in MAPE (0.54) and sMAPE (0.28). Despite this slight deterioration, the hybrid model continues to outperform both XGBoost and SARIMAX in several aspects, consolidating itself as the most robust and versatile option overall.

6. Conclusions and future work

This study provided a comparative assessment of three distinct modeling approaches for wind energy forecasting using SCADA data: a classical statistical model (SARIMAX), a machine learning model (XGBoost), and an advanced hybrid model integrating EMD filter, an attention-enabled LSTM network, and XGBoost. The evaluation was conducted across multiple data scenarios with varying sampling rates and inclusion of key exogenous inputs (wind speed measurements and the turbine's power curve) to mimic real operational data richness. The hybrid model achieved the highest predictive accuracy, outperforming both the SARIMAX and standalone XGBoost approaches. This superiority is attributed to the hybrid model's ability to capture a wider range of patterns: EMD filters noise and non-stationary fluctuations, the LSTM component learns complex temporal dependencies, and XGBoost fine-tunes the final predictions. These findings align with literature suggesting that combining statistical and machine learning methods leverages complementary strengths for improved wind forecasting accuracy [10]. Beyond performance advantages, the hybrid architecture provides multiple interpretability pathways valuable for operational deployment. The EMD decomposition enables temporal scale analysis across frequency bands, the LSTM attention mechanism offers insights into critical forecasting time steps, and the XGBoost component supports standard interpretability tools including feature importance and SHAP analysis to quantify contributions of both exogenous inputs and learned temporal features.

The present results reinforce that a thoughtfully designed hybrid framework can deliver significant accuracy gains over single-model baselines in wind energy forecasting. The analysis also sheds light on forecast horizon effects and the role of exogenous inputs. As expected, prediction error grows with longer lead times: short-term forecasts (e.g. one-hour ahead) were markedly more accurate than day-ahead forecasts, reflecting the well-known degradation of accuracy over extended horizons. This deterioration with increasing horizon is attributable to the compounding uncertainty and volatility in wind patterns. Regarding input factors, the inclusion of relevant exogenous variables was found to enhance model performance. A combined examination of data granularity and exogenous variable contributions further contextualizes the above findings. The study emphasizes that data resolution and feature richness jointly influence forecast accuracy. This perspective also aligns with recent work on data-level approximate computing, which shows that judicious choices in sampling, precision scaling, quantization, and feature selection can reduce computational cost while preserving predictive accuracy—supporting the practicality of high-resolution, feature-rich inputs in forecasting pipelines [31]. Using finely granular data (with high sampling frequency) allows the models to capture transient fluctuations that would be obscured in coarse, aggregated data. This observation is in line with Effenberger et al. (2023), who showed that coarse temporal resolutions (e.g. daily or monthly averages) fail to preserve critical wind speed dynamics, whereas using data at hourly or sub-hourly intervals retains essential variability [9]. In our case, the dataset's temporal granularity (on the order of minutes) enabled the LSTM-based model to learn subtle short-term patterns, while the decomposition via EMD mitigated noise inherent in high-frequency signals. The contributions of individual exogenous variables were also evident – each additional input (e.g. temperature or wind direction) explained a portion of variance that pure time-series data could not. This multi-variable approach improved the model's generalization to changing weather conditions. That said, leveraging very high-resolution data with many features introduces considerable complexity in modeling. Thus, an appropriate balance of data granularity and input diversity was essential to attain the observed accuracy improvements.

The comparative study also highlights important computational trade-offs and implications for deployment. There was a noticeable contrast in computational complexity between the simple statistical model and the deep learning hybrid. SARIMAX, being relatively lightweight, offered fast training and prediction with minimal computational overhead. XGBoost, although more complex than SARIMAX, is still efficient and benefited from parallelizable tree-boosting, making it feasible for near real-time use. In contrast, the hybrid model incurred substantially higher computational cost due to its multi-stage pipeline (decomposition and neural network training) and larger number of parameters. Training the LSTM component, in particular, was time-consuming, and the hybrid model also demands more memory and processing power during execution. Additionally, hyperparameter tuning significantly impacts model performance and computational requirements. The hybrid model's numerous hyperparameters (LSTM layers, EMD modes, XGBoost parameters) require careful optimization, often through grid search or Bayesian optimization. While extensive tuning improves accuracy, it substantially increases training time and may lead to overfitting. Simpler models like SARIMAX require minimal hyperparameter adjustment, making them more robust for operational deployment when training data is limited or when rapid model updates are needed. These differences imply that in practical deployments, one must balance the accuracy gains against resource availability and latency requirements [32]. For instance, in a real-time operational setting (such as a wind farm control system or grid dispatch center), the slight accuracy advantage of the hybrid model must be weighed against its longer computation time. If forecasts need to be updated on the order of minutes, a faster model like XGBoost or SARIMAX might be preferable for ensuring timely predictions. On the other hand, for day-ahead planning or scenarios where batch processing is acceptable, the hybrid model's superior accuracy justifies its use.

The deployment of data-driven models in wind turbine power forecasting introduces key ethical and operational concerns. Model drift-where performance degrades over time due to changing weather patterns or turbine wear-can lead to inaccurate forecasts, affecting grid stability and energy markets. Data outages (e.g., sensor failures or communication disruptions) further compound risks, as missing inputs may cause erroneous predictions. Additionally, integrating ML models with legacy systems poses challenges, as outdated infrastructure may lack the flexibility to handle real-time adaptive algorithms, increasing the likelihood of operational failures. To mitigate these risks, some mechanisms can be applied. For instance, if a model detects anomalies (e.g., sudden wind gusts or storms), it should switch to conservative heuristics or physics-based simulations to ensure stable power estimates. When detecting these abnormal data inputs, human intervention may be required. For persistent issues like model drift, continuous monitoring and periodic retraining-using fresh operational data-help maintain accuracy. By combining adaptive ML with resilient fail-safes, wind turbine forecasting systems can balance innovation with reliability in dynamic environments.

Looking toward future work, several promising directions can be identified to advance wind power forecasting. First, exploration of cutting-edge deep learning architectures, particularly transformer-based models, is recommended. Transformer models, with their self-attention mechanisms, can capture long-range dependencies in time series data more effectively than traditional recurrent networks. Second, integrating Generative Adversarial Networks (GANs) offers a novel avenue for enhancement. GANs can be used to generate realistic wind scenario data or to refine predictions by learning the underlying distribution of forecast errors. Preliminary work using GAN-based approaches has shown improved accuracy for wind power predictions, especially for day-ahead

forecasts [33]. A GAN-driven framework could complement deterministic models by providing probabilistic scenario forecasts, thus enriching decision-making for grid operators. In addition, incorporating robust uncertainty quantification methodologies, including prediction intervals derived from ensemble approaches, conformal prediction techniques, or Bayesian neural networks, would provide critical reliability assessments for wind power forecasts. Such uncertainty measures would enable grid operators to make more informed risk-aware decisions, improve reserve allocation strategies, and enhance the overall reliability and integration of wind energy into power systems. Third, the inclusion of more diverse meteorological and environmental inputs should be pursued. While this study already considered key exogenous variables, future models could incorporate additional high-resolution Numerical Weather Prediction (NWP) outputs and meteorological variables (e.g. pressure maps, temperature profiles aloft, humidity, or turbulence indices). The fusion of data-driven models with physics-based forecasts (such as using NWP guidance as inputs or as a first-guess model) is a promising direction to boost long-horizon forecast reliability. An alternative approach to enhance model validation would be to integrate data from multiple wind farms across diverse terrains (coastal, mountainous, offshore, and plains) and varying climatic conditions to validate model robustness and generalizability. This multi-site validation approach would enable assessment of transferability across different wind regimes, topographical features, and turbine configurations, ultimately leading to more universally applicable forecasting frameworks. Finally, an important area of future development is the real-time integration of these advanced models into operational systems. This involves creating adaptive algorithms capable of online learning – updating model parameters as new data arrives – and ensuring model robustness in the face of streaming data and concept drift. It also requires addressing practical deployment issues such as computational optimization for fast inference, fail-safes for data outages, and user-friendly visualization of forecast results for grid management. Achieving seamless real-time deployment will likely require collaboration between data scientists and power system engineers to test and harden the forecasting tools in live environments.

CRediT authorship contribution statement

Leslie Ricardo de la Rosa: Writing – original draft, Software, Methodology, Conceptualization. **Lía García-Pérez:** Writing – review & editing, Validation. **Matilde Santos Peñas:** Writing – review & editing, Validation, Supervision. **Alejandro Gómez:** Validation, Software.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work has been partially supported by the Spanish Ministry of Science and Innovation under the MCI/AEI/FEDER projects PID2021-123543OBC21 and PID2024-155653OB-C21.

Data availability

It is specified in the paper. The dataset used in this study comes from the Kelmarsh wind farm, located in the UK, and is publicly available through Zenodo [18].

References

[1] R. Asghar, M.J. Anwar, H. Wadood, H. Saleem, N. Rasul, Z. Ullah, Promising features of wind energy: a glance overview, in: 2023 4th International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), IEEE, 2023, pp. 1–6.

[2] S.D. Ahmed, F.S. Al-Ismail, M. Shafiullah, F.A. Al-Sulaiman, I.M. El-Amin, Grid integration challenges of wind energy: a review, IEEE Access 8 (2020) 10857–10878.

[3] R. Pandit, D. Astolfi, J. Hong, D. Infield, M. Santos, Scada data for wind turbine data-driven condition/performance monitoring: a review on state-of-art, challenges and future trends, Wind Eng. 47 (2) (2023) 422–441.

[4] H. Yun, C.D. Giurcăneanu, G. Dobbie, Several approaches for the prediction of the operating modes of a wind turbine, Electronics 13 (8) (2024) 1504.

[5] B. Loza, L.I. Minchala, D. Ochoa-Correa, S. Martinez, Grid-friendly integration of wind energy: a review of power forecasting and frequency control techniques, Sustainability 16 (21) (2024) 9535.

[6] O. Benzohra, S.S. Echcharqaouy, F. Fraija, D. Saifaoui, Integrating wind energy into the power grid: impact and solutions, Mater. Today Proc. 30 (2020) 987–992.

[7] M. Sacie, M. Santos, R. López, R. Pandit, Use of state-of-art machine learning technologies for forecasting offshore wind speed, wave and misalignment to improve wind turbine performance, J. Mar. Sci. Eng. 10 (7) (2022) 938.

[8] J.-Y. Hsu, Y.-F. Wang, K.-C. Lin, M.-Y. Chen, J.H.-Y. Hsu, Wind turbine fault diagnosis and predictive maintenance through statistical process control and machine learning, IEEE Access 8 (2020) 23427–23439.

[9] N. Effenberger, N. Ludwig, R.H. White, Mind the (spectral) gap: how the temporal resolution of wind data affects multi-decadal wind power forecasts, Environ. Res. Lett. 19 (1) (2023) 014015, https://doi.org/10.1088/1748-9326/ad0bd6.

[10] S. Hanifi, X. Liu, Z. Lin, S. Lotfian, A critical review of wind power forecasting methods—past, present and future, Energies 13 (15) (2020), https://doi.org/10.3390/en13153764.

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

[12] A.A. Abdelhamid, E.-S.M. El-Kenawy, A. Ibrahim, M.M. Eid, D.S. Khafaga, A.A. Alhussan, S. Mirjalili, N. Khodadadi, W.H. Lim, M.Y. Shams, Innovative feature selection method based on hybrid sine cosine and dipper throated optimization algorithms, IEEE Access 11 (2023) 79750–79776, https://doi.org/10.1109/ACCESS.2023.3298955.

[13] M.R. Mohapatra, R. Radhakrishnan, R.M. Shukla, A hybrid approach using arima, Kalman filter and lstm for accurate wind speed forecasting, in: 2023 IEEE International Symposium on Smart Electronic Systems (iSES), 2023, pp. 425–428.

[14] M. Du, Improving lstm neural networks for better short-term wind power predictions, in: 2019 IEEE 2nd International Conference on Renewable Energy and Power Engineering (REPE), 2019, pp. 105–109.

[15] J.A. Rodrigo, J.E. Ortiz, Forecasting series temporales con gradient boosting: skforecast, xgboost, lightgbm y catboost, https://cienciadedatos.net/documentos/py56-forecasting-time-series-with-xgboost, Feb. 2021.

[16] S.M. Abdullah, M. Periyasamy, N.A. Kamaludeen, S. Towfek, R. Marappan, S. Kidambi Raju, A.H. Alharbi, D.S. Khafaga, Optimizing traffic flow in smart cities: soft gru-based recurrent neural networks for enhanced congestion prediction using deep learning, Sustainability 15 (7) (2023) 5949.

[17] F. Harrou, A. Dairi, A. Dorbane, Y. Sun, Enhancing wind power prediction with self-attentive variational autoencoders: a comparative study, Results Eng. 23 (2024) 102504, https://doi.org/10.1016/j.rineng.2024.102504.

[18] J.M. Polito, C. Johnson, D.J. Paul, N.W. Hodnett, Kelmarsh wind farm scada dataset, https://zenodo.org/record/5841834, Jan. 2022.

[19] F.R. Alharbi, D. Csala, A seasonal autoregressive integrated moving average with exogenous factors (sarimax) forecasting model-based time series approach, Inventions 7 (4) (2022) 94.

[20] T. Chen, C. Guestrin, Xgboost: a scalable tree boosting system, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794.

[21] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, arXiv preprint, arXiv:1412.3555, 2014.

[22] M. Ali, D.M. Khan, H.M. Alshanbari, A.A.-A.H. El-Bagoury, Prediction of complex stock market data using an improved hybrid emd-lstm model, Appl. Sci. 13 (3) (2023) 1429.

[23] R. Zhu, Y. Yang, J. Chen, Xgboost and cnn-lstm hybrid model with attention-based stock prediction, in: 2023 IEEE 3rd International Conference on Electronic Technology, Communication and Information (ICETCI), IEEE, 2023, pp. 359–365.

[24] K. Aas, M. Jullum, A. Løland, Explaining individual predictions when features are dependent: more accurate approximations to Shapley values, Artif. Intell. 298 (2021) 103502, https://doi.org/10.1016/j.artint.2021.103502.

[25] M. Lepot, J.-B. Aubin, F.H. Clemens, Interpolation in time series: an introductive overview of existing methods, their performance criteria and uncertainty assessment, Water 9 (10) (2017) 796.

[26] M. Zou, S.Z. Djokic, A review of approaches for the detection and treatment of outliers in processing wind turbine and wind farm measurements, Energies 13 (16) (2020) 4228.

[27] G.E. Box, G.M. Jenkins, G.C. Reinsel, G.M. Ljung, Time Series Analysis: Forecasting and Control, John Wiley & Sons, 2015.

[28] P.J. Brockwell, R.A. Davis, Introduction to Time Series and Forecasting, Springer, 2002.

[29] T. Burton, N. Jenkins, D. Sharpe, E. Bossanyi, Wind Energy Handbook, John Wiley & Sons, 2011.

[30] L.R.D. la Rosa, A. Gómez, M. Santos, L. García-Pérez, Advancing wind energy forecasting by integrating power curves into sarimax models, in: Proceedings of the VI WWWE Workshop on Wind and Marine Energy 2024, Madrid, Spain, 2024.

[31] A.M. Dalloo, A.J. Humaidi, Optimizing machine learning models with data-level approximate computing: the role of diverse sampling, precision scaling, quantization and feature selection strategies, Results Eng. 24 (2024) 103451, https://doi.org/10.1016/j.rineng.2024.103451.

[32] A.S. Tahir, A.M. Abdulazzeez, I.A. Ali, A review on deep learning in wind speed forecasting: techniques and challenges, Int. J. Intell. Syst. Appl. Eng. 12 (4) (2024) 3574–3594.

[33] R. Liu, Y. Song, C. Yuan, D. Wang, P. Xu, Y. Li, Gan-based abrupt weather data augmentation for wind turbine power day-ahead predictions, Energies 16 (21) (2023), https://doi.org/10.3390/en16217250.