Journal of Atmospheric and Solar–Terrestrial Physics 277 (2025) 106629

Contents lists available at ScienceDirect

Journal of Atmospheric and Solar-Terrestrial Physics

journal homepage: www.elsevier.com/locate/jastp

Research Paper

Bayesian-optimised hybrid machine learning model for coastal wind gust prediction in a marine-influenced atmospheric boundary layer

Mohammad Reza Chalak Qazani [a,b,*], Mahmood Al-Bahri [b], Muhammad Zakarya [b], Falah Y.H. Ahmed [b], Amirhossein Mohajerzadeh [b], Saeid Hosseini [b], Mehdi Moayyedian [c], Zoran Najdovski [d], Houshyar Asadi [d]

[a] College of Science and Engineering, James Cook University, Townsville, 4814, QLD, Australia
[b] Faculty of Computing and Information Technology, Sohar University, Sohar, 311, Oman
[c] College of Engineering and Technology, American University of the Middle East, Egaila, 54200, Kuwait
[d] Institute for Intelligent Systems Research and Innovation (IISRI), Deakin University, Waurn Ponds, 3216, VIC, Australia

A R T I C L E I N F O

Handling editor: Costas Varotsos

Keywords:
Atmospheric dynamics
Coastal wind gust prediction
Marine boundary layer
Sea–land breeze interactions
Time-series forecasting
Bayesian optimisation
Multilayer perceptron
Meteorological modelling

A B S T R A C T

Accurate prediction of wind gusts is crucial for applications in aviation, coastal and marine operations, and atmospheric dynamics research. This study presents a novel model combining a Sequencing Block and a Layer Perceptron (MLP) optimised using Bayesian Optimisation (B-MLP) to enhance the precision of coastal atmospheric wind gust forecasts. The model is validated using a 13-year dataset (January 2010 to March 2023) from Muscat International Airport, a coastal site influenced by Gulf of Oman sea–land breeze interactions. The Sequencing Block is designed and developed to capture the optimal arrangement of dataset segmentation using atmospheric and boundary layer parameters, thereby enhancing the model's predictive accuracy. The B-MLP model's efficacy is compared against traditional methods, including Decision Tree (DT) and Support Vector Regression (SVR), demonstrating a substantial enhancement in forecast quality. The B-MLP model achieves a correlation coefficient of 0.817 between actual and forecasted wind gusts, outperforming DT and SVR by notable margins in both accuracy and error reduction. The newly proposed model is validated using a 13-year dataset (January 2010 to March 2023) from Muscat International Airport, a coastal site influenced by Gulf of Oman sea–land breeze interactions, to prove its robustness and applicability on a 1-day ahead prediction horizon. The proposed B-MLP model improves forecast accuracy and offers a scalable solution for atmospheric boundary layer studies, marine safety applications, and real-time meteorological data analysis.

1. Introduction

Air movement from areas of high to low pressure induces a meteorological quantity known as wind speed, which is determined by measuring the speed and direction of air over coastal and inland regions. Forecasting wind gusts is a critical challenge in atmospheric science, with direct implications for coastal and marine safety, infrastructure resilience, and energy systems (Attari et al., 2024; Bhasuru et al., 2022). Accurate predictions are crucial for ensuring safety and optimising operations in these areas. For instance, wind gust forecasts are crucial for maritime navigation, enabling ship operators to avoid hazardous sea conditions and ensure safe port entries and departures (Cutajar et al., 2023). Additionally, wave energy systems rely on accurate wind gust predictions to optimise power generation efficiency and protect equipment from extreme sea conditions (Haripriya et al., 2021). This study proposes a data-driven hybrid forecasting model designed to improve the precision of wind gust prediction, particularly in challenging coastal environments. Mean wind speed is the average wind speed over a specified duration (10 min (Beljaars, 1987)). In contrast, wind gusts are sudden and brief increases in wind speed that typically last less than 3 s and are often accompanied by changes in wind direction and turbulence (Minola et al., 2020). In practical applications, various prediction horizons, including short-term and long-term, are required to meet various criteria (Cai et al., 2020). In wind speed prediction, 0–6 h ahead is known as the short-term horizon, while above 6 h ahead is classified as the long-term horizon (Memarzadeh and Keynia, 2020). Despite ongoing advances, accurately predicting wind gusts remains a challenge due to their nonlinear, intermittent, and regionally variable

* Corresponding author. College of Science and Engineering, James Cook University, Townsville, 4814, QLD, Australia.
E-mail addresses: m.r.chalakqazani@gmail.com, mohamadreza.chalakqazani@jcu.edu.au (M.R. Chalak Qazani).

https://doi.org/10.1016/j.jastp.2025.106629
Received 5 August 2025; Received in revised form 8 September 2025; Accepted 10 September 2025
Available online 14 September 2025
1364-6826/

characteristics, especially in coastal environments influenced by complex air–sea interactions. Mathematical models have traditionally been used to predict wind gusts, such as the gust prediction model by Nehama and Reason (2021). In the broader context of atmospheric and space weather, real-time forecasting efforts have also been explored for solar wind dynamics, as discussed by (Fry et al., 2007), which outlined the capabilities and limitations of operational prediction systems. These studies highlight parallels between solar wind and terrestrial wind gust forecasting, particularly in the challenges of handling dynamic, multi-scale processes and the need for robust, data-driven models.

Inaccurate wind gust predictions can have a profound impact on maritime operations, underscoring the critical need for precise forecasting. In maritime operations, particularly in coastal regions and harbours, wind gusts can impact ship handling, docking maneuvers, and the safety of passengers and cargo. Accurate wind forecasting is essential for operational efficiency and the safety of personnel and equipment. Improving forecasting accuracy is thus not merely a technical challenge, but a crucial safety measure that influences everything from flight schedules to shipping routes and, ultimately, the economic and operational stability of these industries.

Various methods have been employed in the literature to enhance the accuracy of wind speed forecasting. For example, Shi et al. (2012) compared the accuracy of the ARIMA-ANN and ARIMA-support vector regression (SVR) hybrid models with the ARIMA, ANN, and SVR single models. They (Shi et al., 2012) concluded that these hybrid models did not always exhibit superior performance in the selected forecasting horizons. Signal decomposition techniques are used in hybrid approaches to decompose the wind speed into sub-series. Recent advances in hybrid numerical models have further enhanced forecasting accuracy by integrating physical simulations with statistical or machine learning components. For example (Brabec et al., 2021), demonstrated that coupling numerical weather prediction models with machine learning post-processing significantly improves wind speed forecasts, especially in complex meteorological scenarios. A study (Di et al., 2019) optimised a mesoscale numerical weather prediction model for wind-speed forecasting using a surrogate-based automatic optimisation method. They employed a multivariate adaptive regression spline for sensitivity analysis and adaptive surrogate modelling-based optimisation for tuning parameters, resulting in improved simulation accuracy and interpretable forecasts. Another study (Tateo et al., 2019) introduced a statistical method based on an ensemble PDF to predict "Wind Days." It outperformed traditional approaches, showing high accuracy in wind speed forecasts by considering errors and estimating the probability of occurrence. Prieto-Herráez et al. (2021) improved wind speed forecasts in complex terrain by combining weather research and forecasting and HDWind models. This two-step approach enhanced accuracy by providing high-resolution wind fields and localised wind speed and direction values, as demonstrated in an urban interface area in Spain. Pan et al. (2021) evaluated a high-resolution weather research and forecasting model with real-time data assimilation for surface wind forecasts over China. The model demonstrated advantages in forecasting detailed weather features but exhibited biases related to wind regimes and topography, particularly during diurnal changes in the summer. Zouaidia et al. (2021) introduced a novel hybrid model for one-day ahead wind speed forecasting in wind farms. It optimises hyperparameters and outperforms benchmark models in accuracy. Gupta et al. (2022) deployed a hybrid machine learning model for short-term wind speed prediction, demonstrating that LDMR achieves the highest prediction accuracy while ELM offers faster computational performance.

Xu and Wei (2022) used LSTM and CNN for ultra-short-term wind speed forecasting, which is crucial for high-speed rail safety. It achieves impressive prediction accuracy, with mean absolute errors of 0.487 m/s for 1 min ahead, 0.547 m/s for 5 min ahead, and 0.593 m/s for 10 min ahead, demonstrating superior performance compared to other models. Suman and Kumar (2023) implemented state-of-the-art machine learning techniques to analyse and mitigate Distributed Denial of Service (DDoS) attacks in cloud computing environments. Jiang et al. (2023) proposed a comprehensive wind speed forecasting system that utilises secondary decomposition, double fuzzy information granulation, CNN-gated recurrent unit hybrid forecasting, and the slime mould algorithm. The system achieves improved deterministic and probabilistic forecasting by capturing trends and uncertainty, enhancing wind power grid risk assessment. Shang et al. (2023) introduced a wind speed forecasting model that utilises ensemble empirical mode decomposition, an attention mechanism, and a CNN to accurately capture implicit wind speed data features, resulting in improved forecasting performance compared to other models. Xing et al. (2024) presented a novel short-term wind speed forecast model integrating attention mechanisms with LSTM, enhancing forecast accuracy and outperforming traditional models, particularly at longer forecast horizons. The LSTM-transformer model has also been successfully applied to predict and analyse sea surface temperatures by Fu et al. (Iqra et al., 2025), highlighting the potential of deep sequence models in capturing temporal dependencies for marine environmental forecasting. Recent research by Iqra et al. (2025) applied Graph Neural Networks for wave prediction at Darwin Harbour, Australia, demonstrating the effectiveness of deep learning in capturing complex spatial-temporal dynamics in coastal marine environments.

This study addresses a critical gap in current forecasting models, which often cannot accurately capture the temporal dynamics and nonlinearity of wind gusts in coastal zones. Traditional models often struggle with unpredictable and intermittent wind gusts, leading to less reliable predictions. To overcome these limitations, we introduce a novel hybrid forecasting model that integrates block sequencing with a Bayesian-optimised multilayer perceptron (B-MLP), designed to enhance temporal learning and parameter selection simultaneously. This approach specifically targets the challenge of hyperparameter optimisation in machine learning models, providing a more precise and efficient framework for wind gust forecasting. This study's key concepts include wind gust forecasting, which refers to predicting sudden and short-duration increases in wind speed, a critical aspect for various safety and operational applications. Machine learning models, such as MLP, decision trees (DT), and SVR, are employed to analyse complex datasets and improve prediction accuracy. Bayesian optimisation is utilised to fine-tune these models by selecting optimal hyperparameters, enhancing the models' predictive performance and efficiency. These concepts are foundational for developing advanced forecasting techniques that address the challenges of accurately predicting wind gusts.

Building on prior MLP-based approaches, this study develops a hybrid forecasting framework that leverages block sequencing and Bayesian optimisation to enhance both model structure and predictive accuracy. The process includes.

1. Data preprocessing to prepare the dataset.
2. This study has utilised Bayesian optimisation to design the sequencing block and determine the optimal MLP structure.
3. Training the MLP using the optimised sequencing block and hyperparameters.

The Bayesian optimiser plays a crucial role in selecting the optimal number of delays, weights, layers, neurons, and other parameters, ultimately leading to a more efficient model. This methodology is applied to a coastal wind dataset collected near the Gulf of Oman, where complex marine-atmospheric interactions significantly influence wind dynamics, demonstrating the model's effectiveness in challenging real-world coastal environments. First, we employ a global optimisation technique (Bayesian) to select the block sequencing of the MLP. Second, we select the hyperparameters of the MLP using the Bayesian optimisation method. Third, we develop a modular model to search for the optimum results by switching between deep learning and machine learning areas, supporting up to 100 layers and 400 neurons in each layer. Bayesian optimisation was selected for this study because it has a vast search space in which to find the optimal MLP hyperparameters. The resulting MLP algorithm with Bayesian optimisation, referred to as B-MLP throughout this paper, yields better results than methods such as DT and SVR or the traditional MLP. Indeed, Bayesian optimisation improves the efficiency of the existing MLP approaches in the literature by selecting optimal hyperparameters. This improvement is primarily due to Bayesian optimisation's ability to efficiently explore the hyper-parameter space, enabling the selection of model configurations that minimise prediction error and enhance generalisation. Additionally, extracting the optimal sequencing block is the second reason behind the higher efficiency of the proposed B-MLP, resulting in the least prediction error. Specifically, we present an application focusing on the time series of daily coastal wind gusts near the Gulf of Oman (recorded at Muscat International Airport from January 2010 to March 2023) to demonstrate the high prediction performance of the proposed B-MLP in a marine-influenced environment.

In summary, this study addresses the persistent challenge of accurately forecasting wind gusts in coastal regions by introducing a novel hybrid machine learning model. While existing models often struggle with the intermittent and dynamic nature of wind gusts, especially in marine-influenced boundary layers, our approach integrates block sequencing with a Bayesian-optimised multilayer perceptron (B-MLP) to capture temporal dependencies and enhance predictive performance. The primary objective of this study is to design and validate a robust, data-driven forecasting framework that improves accuracy over traditional models such as Decision Trees and SVR. By applying this framework to a 13-year coastal wind dataset from the Gulf of Oman, we demonstrate its practical effectiveness and contribution to both meteorological forecasting and atmospheric modelling research.

The paper is organised as follows. Section 2 describes the proposed B-MLP methodology, as well as the DT and SVR approaches we use as benchmarks. In Section 3, we carry out the empirical analysis and validation. Finally, in Section 4, we conclude the remarks of the current study.

2. Methodology

The proposed hybrid model integrates advanced techniques in block sequencing, MLP, and Bayesian optimisation to significantly improve the accuracy of wind gust forecasting. This comprehensive methodology, depicted in Fig. 1(a) and (b), encompasses detailed data preprocessing and model optimisation stages to ensure the model's robustness and precision.

1. **Data Preprocessing and Block Sequencing Design:** The initial step involves meticulous data preprocessing, which is crucial for handling the complexity of the meteorological dataset. This process includes data cleaning, normalisation, and extracting relevant features, which are pivotal for enhancing the model's performance. The dataset comprises multiple meteorological parameters, such as temperature, humidity, and historical wind speed records, which are normalised to ensure consistent data scaling, facilitating the efficient training of the MLP. Block sequencing is integral to the model, capturing the temporal dependencies in the data. This technique involves organising the data into blocks, each consisting of multiple sequential observations. Bayesian optimisation plays a critical role here, as it determines the optimal number of delays within these blocks—essentially, the number of past data points considered in predicting future values. This optimisation ensures the model accurately captures the temporal patterns inherent in wind gust occurrences, as illustrated in Fig. 1(a).

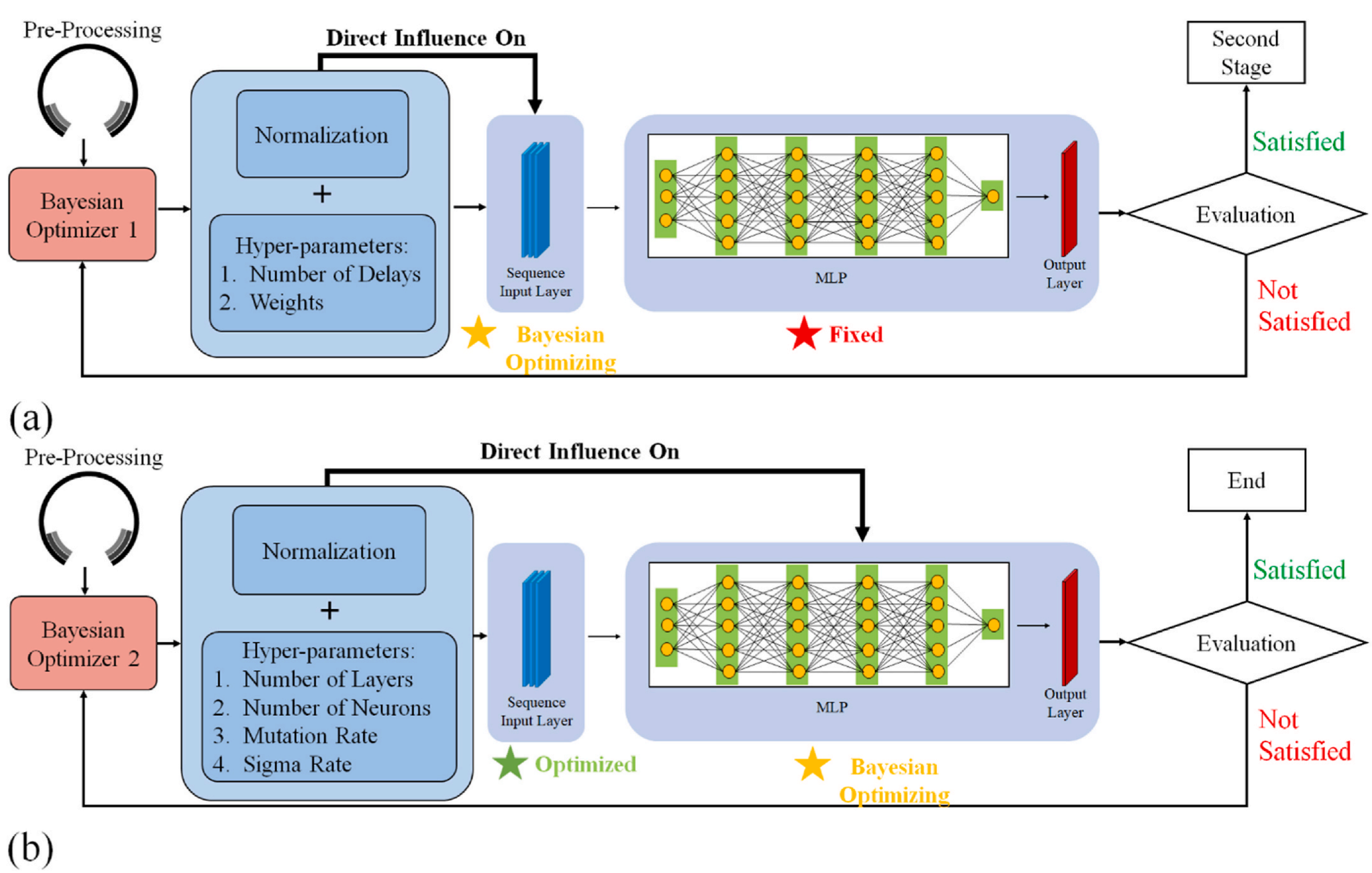

Fig. 1. The proposed hybrid model for predicting the wind gust, including data preprocessing, Bayesian optimisation method, block sequencing, and MLP (a): optimising the block sequencing; (b): optimising the MLP structure.

2. **Optimisation of the MLP Architecture:** Once the block sequencing is established, the focus shifts to optimising the MLP's architecture. The MLP, a type of neural network, is chosen for its capability to model complex, nonlinear relationships in data. The architecture of the MLP, including the number of layers and the number of neurons per layer, has a significant influence on the model's predictive power and computational efficiency. Bayesian optimisation is employed to fine-tune these architectural parameters. Unlike traditional methods that rely on manual or grid search techniques, Bayesian optimisation offers a probabilistic approach, efficiently navigating the parameter space to identify the configuration that minimises the model's prediction error. Specifically, it adjusts the mutation rate and sigma parameters, which control the model's learning rate and the variance in weight updates. This optimisation is crucial for preventing overfitting and ensuring the model generalises well to unseen data, as shown in Fig. 1(b).
3. **Model Training and Fine-Tuning:** The model training process involves an iterative approach. The neural network is trained on the preprocessed dataset, and the mean square error (MSE) between the predicted and actual wind gust values is computed. This metric serves as a key indicator of the model's accuracy. The Bayesian optimiser iteratively refines the block sequencing and MLP architecture parameters to minimise the MSE. This fine-tuning process ensures that the model learns the data's inherent patterns and adjusts to nuances that may not be immediately apparent in the raw data. The sequencing block, once optimised, is defined as the static component of the model. This block, characterised by the optimal number of delays and corresponding weights, effectively encapsulates the influence of historical data on current predictions. The dynamic component, represented by the MLP, is implemented using MATLAB software, providing a flexible and powerful framework for neural network development. The final model incorporates the optimally tuned parameters, ensuring a balanced trade-off between complexity and generalisation capability.
4. **Comparative Analysis and Applications:** In addition to the detailed description of the proposed model, the subsequent sections provide a comprehensive analysis of the dataset, the preprocessing steps, and the specific configuration of the Bayesian optimisation and MLP components. A comparative analysis with traditional models, such as DT and SVR, is also presented. This comparison underscores the superiority of the proposed hybrid model in terms of accuracy and computational efficiency, highlighting its practical applicability in real-world wind gust forecasting scenarios.

Overall, the proposed model represents a significant advancement in wind gust forecasting, offering a sophisticated approach that integrates state-of-the-art machine learning techniques with robust optimisation strategies. The detailed explanation and methodological rigour aim to address the reviewer's concerns and provide a clear and comprehensive understanding of the model's development and potential applications.

In the following sections, we describe the dataset used in the study, the preprocessing procedure adopted, the sequencing block, the applied Bayesian optimisation method, and the MLP model employed. Additionally, we provide the details of the DT and VR models used for comparison purposes.

2.1. Datasets acquisition

In this study, we perform an empirical analysis to forecast coastal wind gusts near the Gulf of Oman, using data recorded at Muscat International Airport as a representative marine-influenced site. We will incorporate external factors, such as the average/variation of temperature, average/variation of humidity, and average wind speed, to enhance the accuracy of the predictor unit. We will also incorporate real-time meteorological information. Wind gust prediction is a complex task influenced by various meteorological parameters. Our study focuses on the following key parameters, selected for their direct impact on wind behaviour.

1. **Temperature:** Temperature gradients contribute to atmospheric pressure differences, which drive wind movement. Temperature variations can affect thermal wind, impacting gust intensity and frequency.
2. **Pressure:** Barometric pressure is a crucial factor in predicting wind patterns. High-pressure systems typically indicate stable weather, while low-pressure systems are associated with more dynamic and often stronger winds, including gusts.
3. **Humidity:** Moisture content in the air, or humidity, affects air density and stability. High humidity can lead to the formation of low-level clouds and fog, which may dampen wind speeds, whereas dry conditions can enhance wind activity.
4. **Wind Direction:** The direction from which the wind is blowing is vital for understanding local wind phenomena and potential obstacles that could alter wind paths, thereby influencing gust occurrences.

These parameters were chosen based on their established significance in meteorological studies and their proven correlation with wind gust phenomena. By incorporating these variables, our model aims to achieve a more accurate and nuanced prediction of wind gusts, which is essential for applications in aviation, marine operations, and renewable energy systems.

The Civil Aviation Authority in Oman provided the dataset used in the analysis. The format of the dataset may vary depending on how it was provided, such as CSV, Excel, Google Sheets, SQL, or NoSQL. Wind gust measurements at Muscat International Airport are recorded using an automated weather station (AWS) operated by the Civil Aviation Authority of Oman. The primary instrument for wind speed and gust detection is a cup anemometer (likely compliant with WMO standards), mounted at a standard height of 10 m. The system reports maximum 3-s gusts over 10-min intervals. The measurement accuracy is approximately ± 0.5 m/s for mean wind speed and ± 1 m/s for gusts, with a resolution of 0.1 m/s. Limitations include susceptibility to turbulence near buildings or terrain-induced effects. In contrast, advantages include continuous, long-term data availability with high temporal resolution suitable for both operational forecasting and model validation. These characteristics support the reliability of the 13-year dataset used in this study. We will specifically focus on the daily time series data from January 2010 to March 2023, which comprises 4840 observations.

A portion of the dataset from 2010 to 2012 is shown in Fig. 2, displaying the time series of the weather parameters, including temperature, humidity, and wind speed. In addition, Table 1 displays a sample of the dataset, including the mean/maximum/minimum values of temperature, relative humidity, sustainability, and wind gust, for the period between September 1st and 10th, 2015.

The dataset provided by the Civil Aviation Authority includes weather-related measurements—temperature (in degrees Celsius), humidity (percentage of water vapour in the air), and wind conditions (wind speed in miles per hour)—recorded between 2010 and 2012 at Muscat International Airport, a coastal site influenced by marine atmospheric dynamics near the Gulf of Oman.

The dataset will likely be labelled since the weather conditions were measured and recorded at specific times and locations. Each data point should have a timestamp and geographic location associated with it. This dataset has numerous potential applications, including weather forecasting, climate analysis, and aviation safety. Through weather data analysis, we can identify patterns and trends in the weather. These patterns and trends can be used to forecast future weather conditions and plan accordingly. Additionally, the weather data can be used to evaluate the safety of aviation operations at the airport, such as takeoffs and landings, under different weather conditions.

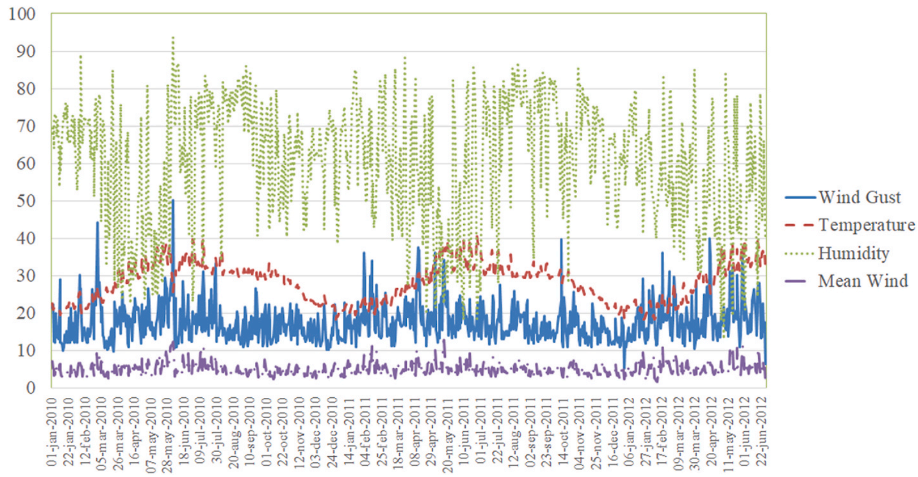

Fig. 2. Dataset representing coastal weather conditions—temperature, humidity, and wind—recorded from 2010 to 2012 at Muscat International Airport, located near the Gulf of Oman and influenced by sea–land breeze dynamics.

Table 1
The descriptive statistics of the data.

Date	Air temperature (°C)			Relative humidity (%)			Wind speed (miles/hour)	
	Mean	Max	Min	Mean	Max	Min	Mean speed	Max gust
01-sep-2015	33.3	39.1	29.8	69	93	44	6	17
02-sep-2015	32.6	36.4	29.1	76	91	44	7	26
03-sep-2015	31.8	34.6	30.3	84	94	74	7	17
04-sep-2015	30.8	33.6	25.7	77	91	58	8	43
05-sep-2015	29.5	31.7	27.7	79	89	69	5	17
06-sep-2015	31.5	36.9	27.6	67	86	42	4	14
07-sep-2015	34.8	39.6	29.5	50	79	34	5	14
08-sep-2015	35.7	40.5	33.9	42	68	29	7	19
09-sep-2015	34.8	37.6	32.0	46	65	28	7	21
10-sep-2015	34.1	37.6	30.9	54	76	33	5	17

2.2. Data preprocessing

In this study, a comprehensive data preprocessing pipeline was implemented to ensure the quality and reliability of the input data, which is critical for the accuracy of the wind gust forecasting model. The key steps involved are as follows.

1. **Data Cleaning and Handling Missing Values:** The dataset included weather-related data such as temperature, humidity, and wind speed. Initially, the dataset was checked for missing or incomplete records, which were addressed using imputation techniques. Missing values were replaced with the mean of the available data to maintain consistency without introducing bias.
2. **Outlier Detection and Removal:** Outliers that could potentially skew the model's learning process were identified and removed. Statistical methods, such as Z-scores and the Interquartile Range (IQR), were used to detect outliers, ensuring that all data points fell within a plausible range for weather conditions.
3. **Normalisation:** Normalisation was applied to ensure that the features contributed equally to the model. This step involved scaling the data to a range of [0, 1] using the Min-Max scaling method. Normalisation is crucial for the MLP model, as it improves convergence during training by preventing any single feature from dominating the learning process.
4. **Feature Extraction and Selection:** Relevant features were extracted from the raw data, including temporal features such as the day of the year and hour of the day, which are significant in capturing seasonal and diurnal patterns in wind gust behaviour. Additionally, features were selected based on their correlation with wind gusts, ensuring that only the most predictive variables were included in the model.
5. **Temporal Sequencing:** Given the time-series nature of the data, a sequencing block was implemented to capture the temporal dependencies between past and future observations. This involved creating lagged variables for key features, which helped the model understand the influence of previous conditions on future wind gust events.
6. **Data Splitting:** The dataset was then divided into training, validation, and test sets. This step was crucial for evaluating the model's performance and ensuring it generalises well to unseen data. The training set was used to fit the model, the validation set was used to tune hyperparameters, and the test set was used to assess the model's final performance.

2.3. Sequencing block

The standard MLP was not designed for time-series data but for curve-fitting purposes. Therefore, it cannot be directly used to analyse time-series data, such as the realised wind gust in this study. The primary motivation for using the Sequencing Block is to facilitate the analysis of temporal patterns in time-series data, such as daily wind gusts. In contrast to traditional methods, such as ARIMA or LSTM networks, the Sequencing Block offers a simpler yet highly effective approach to time-series forecasting.

1. **Simplicity and Computational Efficiency:** The Sequencing Block leverages the strength of MLPs while incorporating temporal information through delays and weights. This approach is computationally less intensive than deep learning models, such as LSTM. It is suitable for applications where computational resources are limited or rapid deployment is required.
2. **Flexibility in Capturing Temporal Dependencies:** The method involves two key hyperparameters: the number of delays and the associated weights, which dictate how past values influence the current value of a variable (e.g., wind gust). These hyperparameters

are fine-tuned using Bayesian optimisation, allowing the model to adaptively capture complex temporal patterns without requiring extensive manual tuning.

3. **Robustness and Generalisation:** Unlike ARIMA, which requires stationarity in the data and can be sensitive to noise, the Sequencing Block approach, combined with Bayesian optimisation, enhances the robustness of the model. It can generalise well across different datasets, as demonstrated by our case study on wind gust forecasting at Muscat International Airport.

We combine the MLP with a sequencing block to handle the time-series data. Two hyperparameters are related to this procedure, namely, the number of delays and weights, which specify how the current value of the wind gust depends on the previous days' wind gusts. We use an input matrix to predict the wind gust on a specific day, and the number of delays is determined using the Bayesian optimiser, as shown in Fig. 1(a). The optimiser is targeted to minimise the MSE between the actual and predicted testing data.

$$\widehat{\boldsymbol{x}}_i = [\,\widehat{\boldsymbol{x}}_{i-d} \quad \cdots \quad \widehat{\boldsymbol{x}}_{i-1} \quad \widehat{\boldsymbol{x}}_i\,] \tag{1}$$

where $\widehat{\mathbf{x}}_i$ denotes the input vector on the day $\boldsymbol{i}$ and $\boldsymbol{d}$ represents the number of delays.

Additionally, we incorporate weight to emphasise the importance of past data relative to current data. Specifically, we adhere to the parsimony principle and use a single weight, denoted as $\boldsymbol{w}$. We modify Eq. (1) slightly using the weight parameter to increase the sensitivity of the dataset for the network as follows:

$$\widehat{\boldsymbol{x}}_i = \left[\,\boldsymbol{w}^d\widehat{\boldsymbol{x}}_{i-d} \quad \cdots \quad \boldsymbol{w}^1\widehat{\boldsymbol{x}}_{i-1} \quad \boldsymbol{w}^0\widehat{\boldsymbol{x}}_i\,\right] \tag{2}$$

The two hyperparameters, $\boldsymbol{d}$ and $\boldsymbol{w}$ are obtained using the Bayesian optimisation method for sequencing. Additionally, we incorporate the external states provided in Fig. 2 and Table 2 into Eq. (2) to reduce the system's complexity in predicting wind gusts using the proposed method. The modified equation is as follows:

$$\widehat{\boldsymbol{x}}_i = \begin{bmatrix} \boldsymbol{w}^d \times \widehat{\boldsymbol{x}}_{i-d} & \cdots & \boldsymbol{w}^1 \times \widehat{\boldsymbol{x}}_{i-1} & \boldsymbol{w}^0\widehat{\boldsymbol{x}}_i \\ \boldsymbol{w}^d \times \mathbf{Ave}(\boldsymbol{T}_{i-d}) & \cdots & \boldsymbol{w}^1 \times \mathbf{Ave}(\boldsymbol{T}_{i-1}) & \boldsymbol{w}^0 \times \mathbf{Ave}(\boldsymbol{T}_i) \\ \boldsymbol{w}^d \times \mathbf{Var}(\boldsymbol{T}_{i-d}) & \cdots & \boldsymbol{w}^1 \times \mathbf{Var}(\boldsymbol{T}_{i-1}) & \boldsymbol{w}^0 \times \mathbf{Var}(\boldsymbol{T}_i) \\ \boldsymbol{w}^d \times \mathbf{Ave}(\boldsymbol{H}_{i-d}) & \cdots & \boldsymbol{w}^1 \times \mathbf{Ave}(\boldsymbol{H}_{i-1}) & \boldsymbol{w}^0 \times \mathbf{Ave}(\boldsymbol{H}_i) \\ \boldsymbol{w}^d \times \mathbf{Var}(\boldsymbol{H}_{i-d}) & \cdots & \boldsymbol{w}^1 \times \mathbf{Var}(\boldsymbol{H}_{i-1}) & \boldsymbol{w}^0 \times \mathbf{Var}(\boldsymbol{H}_i) \\ \boldsymbol{w}^d \times \mathbf{Ave}(\boldsymbol{S}_{i-d}) & \cdots & \boldsymbol{w}^1 \times \mathbf{Ave}(\boldsymbol{S}_{i-1}) & \boldsymbol{w}^0 \times \mathbf{Ave}(\boldsymbol{S}_i) \end{bmatrix} \tag{3}$$

where $\boldsymbol{T}$, $\boldsymbol{H}$, and $\boldsymbol{S}$ indicate the temperature (○C), humidity (%), and wind speed (miles/hour), respectively. Also, **Ave** and **Var** are functions to extract the average and variation of the parameters.

A 1-day-ahead prediction of wind gusts is obtained as the output $\boldsymbol{y}_i$ of the neural network as follows:

$$\boldsymbol{y}_i = \widehat{\boldsymbol{x}}_{i+1} \tag{4}$$

In the current study, we consider three different neural network architectures, namely, the MLP, DT, and SVR, which are described as follows.

2.4. Decision tree (DT)

DT is a data mining technique for classification and regression tasks in machine learning. The type of task depends on whether the target variable is categorical or continuous (Myles et al., 2004). DT comprises nodes and lines, with two types of nodes: branch and leaf nodes. The lines represent the decision pathways between parents and children.

The logic of DT involves splitting data based on conditions and passing inputs to its children, using a binary or ternary decision. This process continues until the data reaches the leaf nodes, where the final prediction results are obtained. The most popular DT algorithms are C4.5, classification and regression trees (CART), and Chi-squared automatic interaction detection (CHAID) (Timofeev, 2004; van Diepen and Franses, 2006; Perner et al., 2001).

DT has an advantage over traditional prediction models because it can produce logical statements or interpretable rules that are easy to understand. This study employs the CART model, which uses a nonparametric regression based on recursive partitioning for wind gust forecasting.

The DT is employed in our model primarily for regression tasks, predicting continuous outcomes—in this case, wind gusts. The DT's hierarchical structure allows it to handle nonlinear relationships by recursively partitioning the dataset based on feature values. In our proposed model, the DT takes as input a set of features derived from time-series data, including historical weather data (such as temperature, humidity, and wind speed). Each node in the tree represents a decision based on one of these features, with the tree's branches leading to further decisions or terminal nodes, which represent the final output or predicted value. The criterion for splitting nodes (e.g., MSE reduction in regression) is chosen to optimise the prediction accuracy of wind gusts. For instance, in predicting wind gusts, the DT model analyses historical patterns in the data and splits the dataset at each node based on the most informative features. This recursive partitioning continues until a stopping criterion is met (such as a minimum number of samples per leaf or a maximum tree depth), resulting in leaf nodes that provide the forecasted wind gust speed. The key advantage of using a DT in our model lies in its interpretability and ability to handle interactions between features without requiring extensive data preprocessing. By visualising the tree, we can derive if-then rules that describe how different weather conditions lead to varying wind gust intensities, which can be particularly useful for decision-making and risk assessment in meteorological applications.

Within the hybrid model framework, the DT complements other models, such as the MLP and SVR, by providing a different approach to capturing patterns in the data. Specifically, the DT's ability to model complex, nonlinear relationships and interactions between features enhances the robustness of the ensemble predictions made by the hybrid model. During the model's operation, predictions from the DT are combined with those from the MLP and SVR to produce a final ensemble forecast that leverages the strengths of each method.

Table 2
Parameters of the SVR, DT, and B-MLP.

Method	Parameters
SVR	Kernel regression = "fitrkernel"; Standardize = "true"
DT	Categorical predictors = 4; Minimum parent size = 5; Maximum number of splits = 5
B-MLP	Number of delays = 2; Weight = 1.3783; Number of layers = 2; Number of neurons = 21; Mutation = 5.00 × 10-3; Sigma = 1.0194 × 10-9

2.5. Support vector regression (SVR)

The SVR is a machine-learning technique introduced by Cortes and Vapnik (Boser et al., 1992). It has gained popularity in meteorology due to its ability to handle both linear and nonlinear time-series data effectively. It falls under supervised learning techniques and can be used for classification and regression tasks. The SVR is particularly suitable for small datasets with a nonlinear structure and can be applied to several applications, including approximation. This study uses an SVR based on polynomial regression for wind gust forecasting. We rearrange the Data for SVR as follows:

$$\boldsymbol{D} = \{(\boldsymbol{x}_1, \boldsymbol{y}_1), (\boldsymbol{x}_2, \boldsymbol{y}_2), \cdots, (\boldsymbol{x}_n, \boldsymbol{y}_n)\},\ \boldsymbol{x}_i \in \mathbb{R}^m,\ \boldsymbol{y}_i \in \mathbb{R}, i = 1, 2, \ldots, n \tag{5}$$

The goal of SVR is to find a regressor function $\boldsymbol{f}(\boldsymbol{x})$ that has a maximum deviation of $\boldsymbol{\varepsilon}$ from the observed targets $\boldsymbol{y}_i$, while also being as flat as possible. This way, errors lower than $\boldsymbol{\varepsilon}$ are acceptable, but deviations more significant than this are not tolerated. We can consider the case where the regression function is a hyperplane:

$$\boldsymbol{f}(\boldsymbol{x}) = \langle \boldsymbol{v}|\boldsymbol{x}\rangle + \boldsymbol{b} \tag{6}$$

where $\boldsymbol{v}$ is the so-called support vector.

To ensure the flatness of the hyperplane, we determine $\boldsymbol{v}$ for solving the following optimisation problem as follows (van Diepen and Franses, 2006):

$$\boldsymbol{min}\frac{1}{2}\|\boldsymbol{v}\|^2 \tag{7}$$

subject to

$$\boldsymbol{y}_i - \langle \boldsymbol{v}|\boldsymbol{x}_i\rangle - \boldsymbol{b} \le \boldsymbol{\varepsilon} + \boldsymbol{\theta}_i \text{ , } \boldsymbol{i} = \mathbf{1}, \mathbf{2}, \ldots, \boldsymbol{n} \tag{8}$$

$$\langle \boldsymbol{v}|\boldsymbol{x}_i\rangle - \boldsymbol{b} - \boldsymbol{y}_i \le \boldsymbol{\varepsilon} + \boldsymbol{\theta}_i^*, \boldsymbol{i} = \mathbf{1}, \mathbf{2}, \ldots, \boldsymbol{n} \tag{9}$$

$$\boldsymbol{\theta}_i \text{ , } \boldsymbol{\theta}_i^* \ge \mathbf{0} \tag{10}$$

The above problem can be solved using the Lagrange multiplier technique.

This study uses SVR as a core component for wind gust forecasting due to its robustness in handling small datasets with nonlinear structures. The polynomial kernel function captures the complex relationships between the input features and the wind gust outputs. This choice of kernel is based on empirical evidence demonstrating its superior performance in terms of prediction accuracy compared to other kernel functions, such as Gaussian or radial basis functions. To train the SVR model, the input data **D** is first preprocessed to normalise the range of values and ensure uniform scaling across features. The model then attempts to find a hyperplane in the transformed feature space that minimises the prediction error while keeping the model complexity low, as defined by the regularisation term. This regularisation helps to prevent overfitting, making the model more generalizable to unseen data. In our integrated model, the SVR is a complementary method alongside the DT and MLP models. The hybrid approach leverages the strengths of each model, with SVR providing a robust solution for capturing nonlinear trends in the wind gust data. The final output of the model is a weighted ensemble of predictions from the SVR, DT, and MLP, where the weights are determined based on the performance of each model during the validation phase.

However, other regressor functions can be used instead of the hyperplane in Eq. (6) to improve the predictive ability during training. The Gaussian kernel, radial basis, and sigmoid functions are the most common regressors. Our empirical application found that the polynomial regressor yields the highest prediction accuracy among all these functions.

2.6. Multilayer perceptron (MLP)

The perceptron neural networks were developed in the 1950s. They were initially used as basic binary classification methods for categorising inputs. At the time, the perceptron was not an algorithm but a machine, and Rosenblatt invented the first perceptron in hardware (Rosenblatt, 1957). MLPs with a single hidden layer are often referred to as "vanilla" (Hastie et al., 2009).

This study uses MLPs with various arrangements of hidden layers, which combine input and output layers. Apart from the input nodes, the neurons use a nonlinear activation function. In the training phase, we obtain the network parameters, such as weights and biases, by minimising the mean absolute error (MAE) between the predicted and realised wind gust. We use a gradient-based optimisation algorithm, specifically stochastic gradient descent, to solve the minimisation problem. We calculate the error gradient with respect to weights and biases using a backpropagation technique. The entire procedure is summarised below.

The activation function can be defined as follows:

$$\boldsymbol{f}(\boldsymbol{x}) = \frac{\mathbf{1}}{\mathbf{1} + \boldsymbol{e}^{-\boldsymbol{x}}} \tag{11}$$

In the first step of the backpropagation method, the error term ($\boldsymbol{\varepsilon}_k^s$) of the kth neuron in the sth layer is calculated as follows:

$$\varepsilon_k^s = \begin{cases} t_k - o_k^s, s = M \\ \sum_{j=1}^{N^{s+1}} w_{jk}^{s+1}\delta_j^{s+1}, s = 1, 2, \ldots, M-1 \end{cases} \tag{12}$$

where $\boldsymbol{M}$ is the number of hidden layers. Also, $\boldsymbol{t_k}$, $\boldsymbol{o}_k^s$, $\boldsymbol{w}_{jk}^{s+1}$, and δ_j^{s+1} are the kth target vector's value, the output of the jth neuron in the sth layer, the synaptic weight of the jth neuron in the (s+1)th layer, and the local gradient of the jth neuron in the (s+1)th layer. $\boldsymbol{N}^{s+1}$ is the number of neurons in the (s+1)th layer.

The local gradient can be calculated as follows:

$$\delta_j^{s+1} = \varepsilon_k^{s+1} f'\left(H_k^{s+1}\right), s = 1, 2, \ldots, M \tag{13}$$

where $\boldsymbol{\varepsilon}_k^{s+1}$ is the error term of the kth neuron in the (s+1)th layer. The derivative of the activation function is shown as $\boldsymbol{f}'\left(\boldsymbol{H}_k^{s+1}\right)$ based on the weighted input sum.

During the gradient descent optimisation process, the weights in the (s+1)th layer are updated as follows:

$$\boldsymbol{w}_{kj}^{s+1} := \boldsymbol{w}_{kj}^{s+1} + \Delta\boldsymbol{w}_{kj}^s + \ldots + \Delta\boldsymbol{w}_{kj}^s \, \eta\delta_k^{s+1}\boldsymbol{o}_j^s, s = 1, 2, \ldots, M \tag{14}$$

where $\boldsymbol{k} = \mathbf{1}, \mathbf{2}, \ldots, \boldsymbol{N}^{s+1}$, $\boldsymbol{j} = \mathbf{1}, \mathbf{2}, \ldots, \boldsymbol{N}^s$ η, and $\boldsymbol{\eta}$ is a free parameter, which is also known as the learning rate.

The MLP in the proposed model is designed to handle time-series data, with a particular focus on forecasting wind gusts. Here's how it works.

1. **Input Representation:** The input data comprises historical wind gust measurements and other meteorological factors, such as temperature and humidity. The data is normalised and structured into blocks containing sequential observations to capture temporal dependencies.
2. **Network Architecture:** The MLP architecture comprises multiple hidden layers, enabling the model to learn complex, nonlinear relationships within the data. Each neuron in these layers uses a nonlinear activation function, specifically the sigmoid function, to introduce nonlinearity into the model. This helps in capturing intricate patterns that linear models might miss.
3. **Training Process:** The MLP is trained using stochastic gradient descent (SGD) to minimise the mean absolute error (MAE) between predicted and actual wind gust values. The backpropagation algorithm calculates the gradients of the error concerning the weights and biases, which are then updated iteratively to minimise the error.
4. **Sequencing Block Integration:** A unique feature of the model is the inclusion of a Sequencing Block, which is tailored to handle the time-series nature of the data. The Sequencing Block incorporates delays and weights that represent the influence of past data on current predictions. This block enables the MLP to consider temporal patterns effectively.
5. **Hyperparameter Optimisation: Bayesian Optimisation is employed to optimis**e the model's hyperparameters, including the number of layers, neurons per layer, delays, and weights. This process ensures that the MLP structure is fine-tuned for maximum predictive accuracy and efficiency, reducing overfitting and enhancing generalisation capabilities.
6. **Output and Prediction:** The final output layer provides the predicted wind gust values. The model's accuracy is assessed by comparing these predictions with the actual data, and the model is refined further if necessary.

2.7. Bayesian Optimisation method

Bayesian optimisation is a global optimisation algorithm that minimises an objective function, which can be either stochastic or deterministic, based on the probability distribution of the system's states (Pelikan et al., 1999). The algorithm utilises Bayes's theorem [40] to handle integers, continuous real values, or categorical states. It is a powerful tool for automatically tuning machine learning models to achieve high accuracy. Since the exact functional form of the objective function is unknown, Bayesian optimisation treats it as a random function. It places a prior distribution over it, typically a Gaussian distribution. The algorithm iteratively updates the posterior distribution over the objective function using function evaluations. The following evaluation point is chosen by constructing an acquisition function based on the objective value.

We apply Bayesian optimisation to accomplish two tasks: optimising the number of delays and weights in the sequencing block, and finding the optimal hyperparameters of the MLP, i.e., the number of layers, the number of neurons, and the learning rate. We search for the optimal parameters of the sequencing block and the optimal hyperparameters of the MLP in a vast domain: the number of delays may be between 1 and 100, the weight may be between 0.5 and 10, the number of layers may be between 1 and 20, the number of units may be between 1 and 400, the mutation rate may be between 5.0 × 10-3 and 7.0 × 10-3, and the sigma rate may be between 1.0 × 10-6 and 1.5 × 10-6.

As the objective function $\boldsymbol{J}$ of the Bayesian optimisation algorithm, we use the MSE between the realised and the predicted wind gust as follows:

$$\boldsymbol{J} = \boldsymbol{MSE}(\boldsymbol{T} - \boldsymbol{P}) \tag{15}$$

where $\boldsymbol{T}$ and $\boldsymbol{P}$ are the targets and predicted output vectors during the training/testing process of the network.

In the Bayesian Optimisation process, the Multilayer Perceptron (MLP) hyperparameters were optimised by exploring a predefined parameter space. The specific ranges for each hyperparameter were: End Delay = 1 to 100, Weight = 0.5 to 10, Number of Layers = 1 to 10, Number of Neurons = 1 to 100, Mu (Mutation Rate) = 0.005 to 0.007, and Sigma = 1e-6 to 1.5e-6. These ranges were selected to cover a broad spectrum of potential configurations, allowing the Bayesian Optimisation algorithm to identify the most effective hyperparameter values for minimising the mean squared error (MSE) during training. The optimisation process involved iterating through different combinations within these ranges, using the bayesopt function in MATLAB to converge on an optimal set of parameters.

3. Results and discussion

We test and compare the DT, SVR, and B-MLP in forecasting the Muscat International Airport wind gust based on the time series dataset from 2010 to 2023. All the numerical experiments are performed using MATLAB software. Specifically, to fit the RT and SVR, we employ the fitrtree.m and fitrsvm.m functions, respectively. Moreover, to implement the Bayesian optimisation method, we use the bayesopt.m function.

3.1. Verification

We tabulate the parameters passed to the fitrtree.m and fitrsvm.m functions (specifying the DT and SVR procedures) in Table 2. This table also reports the optimal block sequencing parameters and the optimal hyperparameters of the B-MLP.

Fig. 3(a–b) illustrates the minimisation process of the fitness function using the proposed Bayesian optimisation unit to extract the optimal sequencing parameters and MLP's hyperparameters, respectively. Fig. 3 (a) shows the extracted mean of the model, observation points, next trial point, and feasible minimum point. It shows that increasing the delay number decreases the model's accuracy. The rate of decrease in model accuracy with respect to the delay number is higher for a lower weight than for a higher weight. The extracted optimal parameters for the sequencing part of the model are 2 for the number of delays and 1.3783 for the weight.

On the other hand, Fig. 3(b) illustrates the fitness function behaviour of the Bayesian optimisation method in determining the optimal hyperparameters of the MLP. It converges to the optimal point after 241 trials with a value of 1.9221. The extracted optimal hyperparameters are 2, 21, 5.00 × 10^{-3}, and 1.0194 × 10^{-9} for the number of hidden layers, number of neurons, mutation rate, and Sigma, respectively.

3.2. Validation

We test the performance of the proposed machine learning approaches using the correlation coefficient between input and output (CC).

$$CC = \frac{\sum_{i=1}^{n}(x_{i+1} - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_{i+1} - \bar{x})^2(y_i - \bar{y})^2}} \tag{16}$$

We also use four error measures, namely, the MSE, the root MSE (RMSE), and the coefficient of determination (R^2):

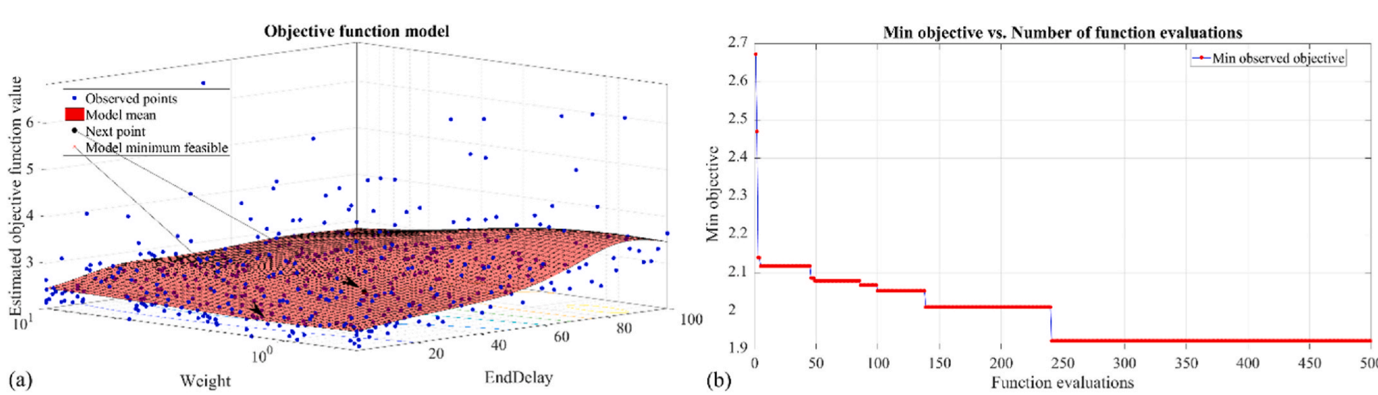

Fig. 3. The fitness function is computed using the Bayesian optimisation iterative procedure. (a): fitness function for the block sequencing parameters; (b): fitness function for the hyperparameters of the proposed B-MLP.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y}_i| \quad (17)$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2 \quad (18)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}{n}} \quad (19)$$

$$R^2 = 1 - \frac{\sum(y_i - \widehat{y}_i)^2}{\sum(y_i - \overline{y})^2} \quad (20)$$

where $\boldsymbol{n}$, $\boldsymbol{x_i}$, $\overline{\boldsymbol{x}}$, $\boldsymbol{y_i}$, $\overline{\boldsymbol{y}}$, and $\widehat{\boldsymbol{y}}_i$ are the number of observations in the considered samples, the *i*th input, the mean of the inputs, the *i*th observed output, the mean of the observed output, and the *i*th predicted value, respectively. It should be noted that the input denotes the actual wind gust. At the same time, the output represents the predicted wind gust using the three investigated methods.

Table 3 shows the results obtained during the training and testing stages, where the performance of the three investigated methods was evaluated. Notably, 70 % of the dataset is used for training (3261 observations) and 30 % for testing (1398 observations). The results show that the SVR algorithm performs poorly among the three investigated methods, with the lowest values of CC and R^2 and the highest values of MSE and RMSE during both the training and testing stages. The DT algorithm is the second most accurate among the investigated models, with correlation CC and R^2 values of 7.52×10^{-1} and 5.66 × 10-1, respectively, during the training stage.

During testing, the DT algorithm achieved a CC and R^2 of 7.77×10^{-1} and 6.04×10^{-1}, respectively. The B-MLP algorithm outperformed the other models, achieving the highest CC and R^2 values of 8.17×10^{-1} and 6.67×10^{-1} during the testing stage. Compared to DT, B-MLP showed improvements of 5.85 % and 11.84 % in CC and R^2, respectively, during the testing stage.

Although achieving high levels of accuracy during the training stage is important, the ultimate goal of machine learning is to obtain accurate predictions during the testing stage. Therefore, the performance of the investigated models was evaluated based on their accuracy during the testing stage, as shown in Table 3. As shown in Table 3, the B-MLP algorithm provides the best performance in the testing stage.

The B-MLP algorithm outperforms the other models in terms of accuracy during the testing stage, with a CC of 8.17×10^{-1}, which is much higher than the CC value provided by the SVR algorithm (3.18×10^{-2}) and 5.14 % higher than the CC value provided by the DT algorithm (7.77×10^{-1}). Furthermore, the B-MLP algorithm achieves an R^2 of 6.67×10^{-1} during the testing stage, which is higher than those of the SVR (0.61×10^{-1}) and DT (0.04×10^{-1}) algorithms. Additionally, the MSE and RMSE of the B-MLP algorithm are consistently much lower than those of the DT and SVR algorithms during the testing stage, as shown in Table 3. The MAE results further support this, with the B-MLP achieving the lowest MAE (2.075) compared to DT (2.259) and SVR (4.170) during testing.

Fig. 4(a–c) shows the error histogram of the three investigated SVR, DT, and B-MLP models during the testing stage. Fig. 4(a) reports the error histogram of the DT during the testing stage, with a mean and standard deviation equal to 2.26 and 2.04 (miles/hour), respectively. The distribution of the error histogram for the DT model exhibits random behaviour, as shown in Fig. 4(a). Fig. 4(b) shows the error histogram of the B-MLP in the testing stage, with mean and standard deviation equal to 4.17 and 4.01 (miles/hour), respectively. The distribution of the error histogram for the SVR model exhibits right-skewed behaviour, as shown in Fig. 4(b). Fig. 4(c) shows the error histogram of the B-MLP in the testing stage, with a mean and standard deviation equal to 2.08 and 1.87 (miles/hour), respectively. The distribution of the error histogram for the B-MLP model exhibits random behaviour, as shown in Fig. 4(c).

Fig. 5(a–c) shows the scatter plot of three investigated models (e.g., SVR, DT, and B-MLP) during the training and testing stages of the models. Fig. 5(a) displays the scatter plot of the DT for both the training and testing stages, with an R-squared value of 0.75272. The DT model is the second most accurate model among the three investigated models for predicting wind gusts. Fig. 5(b) illustrates the scatter plot of the SVR for the training and testing stages with an R^2 of 0.01532. The SVR model is the least accurate among the three investigated models for predicting wind gusts. Fig. 5(c) depicts the scatter plot of the B-MLP for the training and testing stages with an R^2 of 0.79665. The B-MLP model is the most accurate for predicting wind gusts among the three models investigated in this study.

Fig. 6 (a, b, c) shows the actual and predicted 1-day ahead wind gust predictions using DT, SVR, and B-MLP during testing, respectively. Fig. 6 (a) illustrates that the DT model is considerably inaccurate in testing due to the complex and stochastic nature of wind gust time-series data. The model's inability to capture the unpredictable and dynamic behaviour of the wind gusts is evident from the deviation between the predicted and actual wind gust values in the graph. The proposed B-MLP can handle the complex and stochastic trends of the data, as shown in Fig. 6(c), due to the highly efficient sequencing block and Bayesian parameter optimisation. Moreover, Fig. 6(b) shows the actual and predicted 1-day ahead wind gust prediction using SVR with a CC value of 0.0318.

Fig. 7 shows the error between the actual and predicted wind gusts using the three investigated models (DT, SVR, and B-MLP). Based on the presented results, the MSE between the actual and predicted wind gust during the testing process using B-MLP is 76.78 % lower than that of the SVR model. Additionally, the MSE between the actual and predicted wind gusts during the testing process using B-MLP is 16.16 % lower than that of the DT model. Additionally, the RMSE between the actual and predicted wind gusts during the testing process using B-MLP is 51.81 % lower than that of the SVR model. Furthermore, the RMSE between the actual and predicted wind gusts during the testing process using B-MLP is 8.52 % lower than that of the DT model.

To evaluate the statistical significance of the differences in model performance, we conducted paired t-tests on the MAE values derived from 10 independent runs of each model. The tests were performed using MATLAB, as per the code snippet provided below. The MAEs were calculated between the predicted outputs and ground truth values for each model: DT, SVM, and the proposed B-MLP. The results indicate that the difference between DT and MLP was not statistically significant ($p = 0.601$, $h = 0$). In contrast, the differences between MLP and SVM ($p \approx$

Table 3
The results of the three investigated methods (SVR, DT, and B-MLP) are presented.

Parameters		MAE	CC	MSE (miles/hour)	RMSE (miles/hour)
SVR	Training	4.547	8.00×10^{-3}	3.87×10^{1}	6.22
	Testing	4.170	3.18×10^{-2}	3.35×10^{1}	5.79
DT	Training	2.264	7.52×10^{-1}	1.08×10^{1}	3.29
	Testing	2.259	7.77×10^{-1}	9.28	3.05
B-MLP	Training	2.069	7.96×10^{-1}	9.13	3.02
	Testing	2.075	8.17×10^{-1}	7.78	2.79

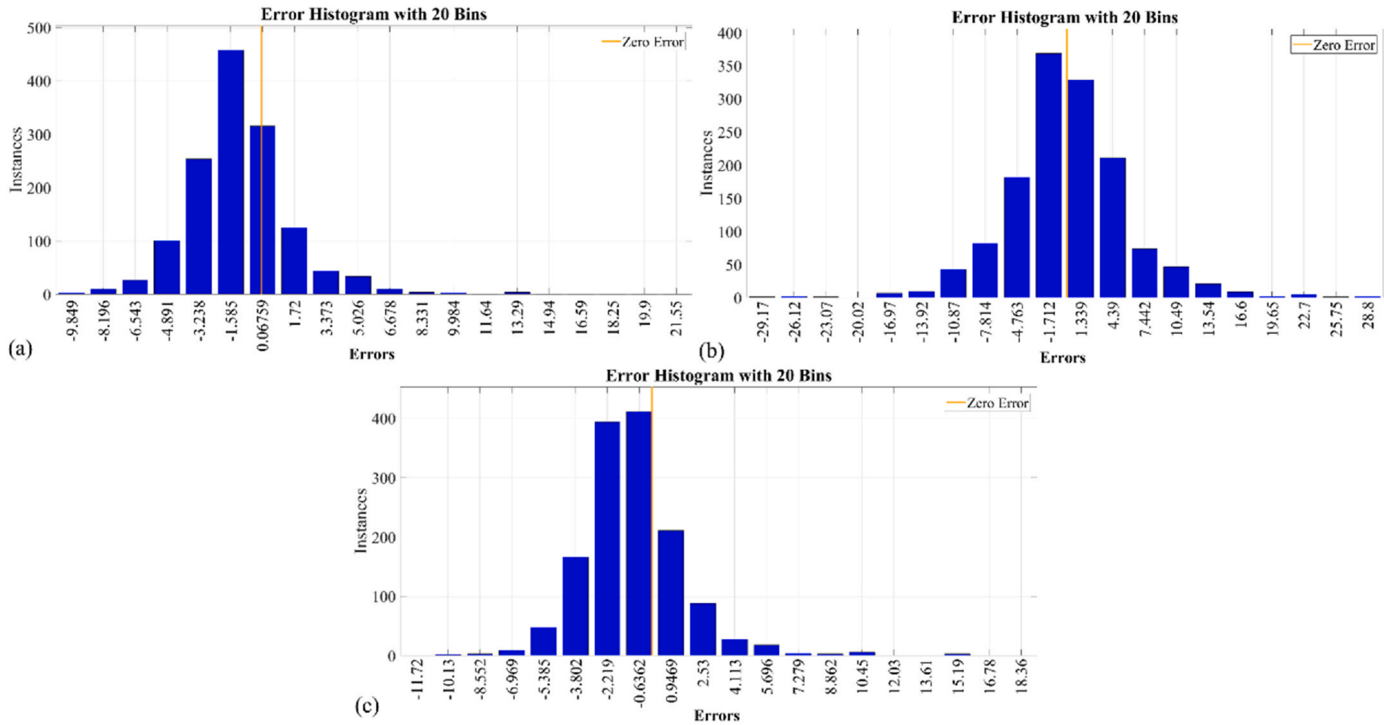

Fig. 4. The error histogram during the testing stage of the proposed investigated models, including (a): DT; (b): SVR; (c): B-MLP, for the prediction of wind gusts.

1.27e-08, h = 1), and between DT and SVM (p = 0, h = 1) were statistically significant at the 95 % confidence level. These findings confirm that the MLP model significantly outperforms SVM. At the same time, its improvement over DT, although numerically evident, is not statistically significant based on the current sample. This statistical validation supports the robustness of the proposed model and adds confidence to the observed prediction gains.

To further assess the reliability of the model performance estimates, 95 % confidence intervals (CIs) were computed for the MAE of each forecasting model using the standard normal approximation. The confidence interval for the DT model was extremely narrow, with a mean MAE of approximately 2.259 and a 95 % CI of [2.2591, 2.2591], indicating highly consistent error across runs. For the SVM, the MAE was significantly higher, with a 95 % CI of [4.1731, 4.1731], showing less variability but overall poorer performance. The proposed B-MLP model yielded a mean MAE within a 95 % CI of [2.1221, 2.5009], reflecting improved accuracy and moderate variation across test runs. These intervals support the conclusion that the MLP model provides a more accurate and robust forecast than SVM and performs comparably to DT, with slightly wider variability due to its more complex structure.

A T-test is performed for the proposed B-MLP model with 1000 trials to evaluate the model's robustness. The extracted results, including the minimum, maximum, and mean of the predicted B-MLP after 1000 trials, are shown in Fig. 8. The represented result shows that the actual wind gusts are always within the range of the predicted values. Additionally, the mean exhibits an excellent fit with the actual wind gust, with CC and R^2 values of 0.83502 and 0.47476, respectively.

In addition to comparing our proposed B-MLP model with traditional machine learning models such as SVR and DT, we also examined results from recent literature employing advanced deep learning architectures, including Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU). For example, Xu and Wei (2022) achieved a Mean Absolute Error (MAE) of 0.487 m/s using LSTM-CNN for 1-min-ahead forecasting. Xing et al. (2024) integrated attention mechanisms into LSTM and reported improved prediction accuracy for longer horizons. Our proposed B-MLP model, designed for 1-day-ahead coastal wind gust prediction, achieved an MAE of approximately 0.927 m/s (2.075 miles/hour) and a correlation coefficient (CC) of 0.817, outperforming SVR and DT. While LSTM/GRU-based methods are powerful for sequence modelling, they often require substantial training time and fine-tuning. In contrast, the B-MLP approach offers a computationally efficient, scalable, and accurate alternative for medium-range wind forecasting, with added interpretability and reduced complexity due to the integrated block sequencing and Bayesian optimisation framework (Table 4).

The results above prove the efficiency of the B-MLP in predicting wind gusts with higher accuracy compared with traditional machine learning methods, including SVR and DT. In addition, an optimised sequencing block is added to the model to decrease the dataset's complexity for the developed B-MLP model. The validation parameters, CC and RMSE, are used to compare the extracted results obtained using three investigated methods, which lead to higher CC and lower RMSE values for B-MLP compared with DT and SVR.

The practical applications of this research are extensive, particularly in fields that require precise predictions of wind gusts. The proposed method enhances flight safety and efficiency for aviation by providing accurate turbulence forecasts. In the renewable energy sector, it helps optimise wind turbine operations, reducing maintenance costs and improving energy output. Additionally, this method can be used in urban planning and construction, where wind effects on structures are a concern. Overall, the proposed method provides a robust tool for decision-making in various wind-sensitive operations.

4. Conclusion

Accurate prediction of wind gusts is critical for ensuring the safety of passengers and crew during operations at airports and seaports. Traditional machine learning methods, such as decision trees (DT) and support vector regression (SVR), are commonly employed to predict time series signals, including wind speed. However, traditional machine learning models such as DT and SVR often suffer from overfitting, limited robustness, sensitivity to small variations, and reduced interpretability. While traditional methods such as DTs or random forests are commonly used for forecasting weather time series signals, their accuracy is often inaccurate for modern international ports.

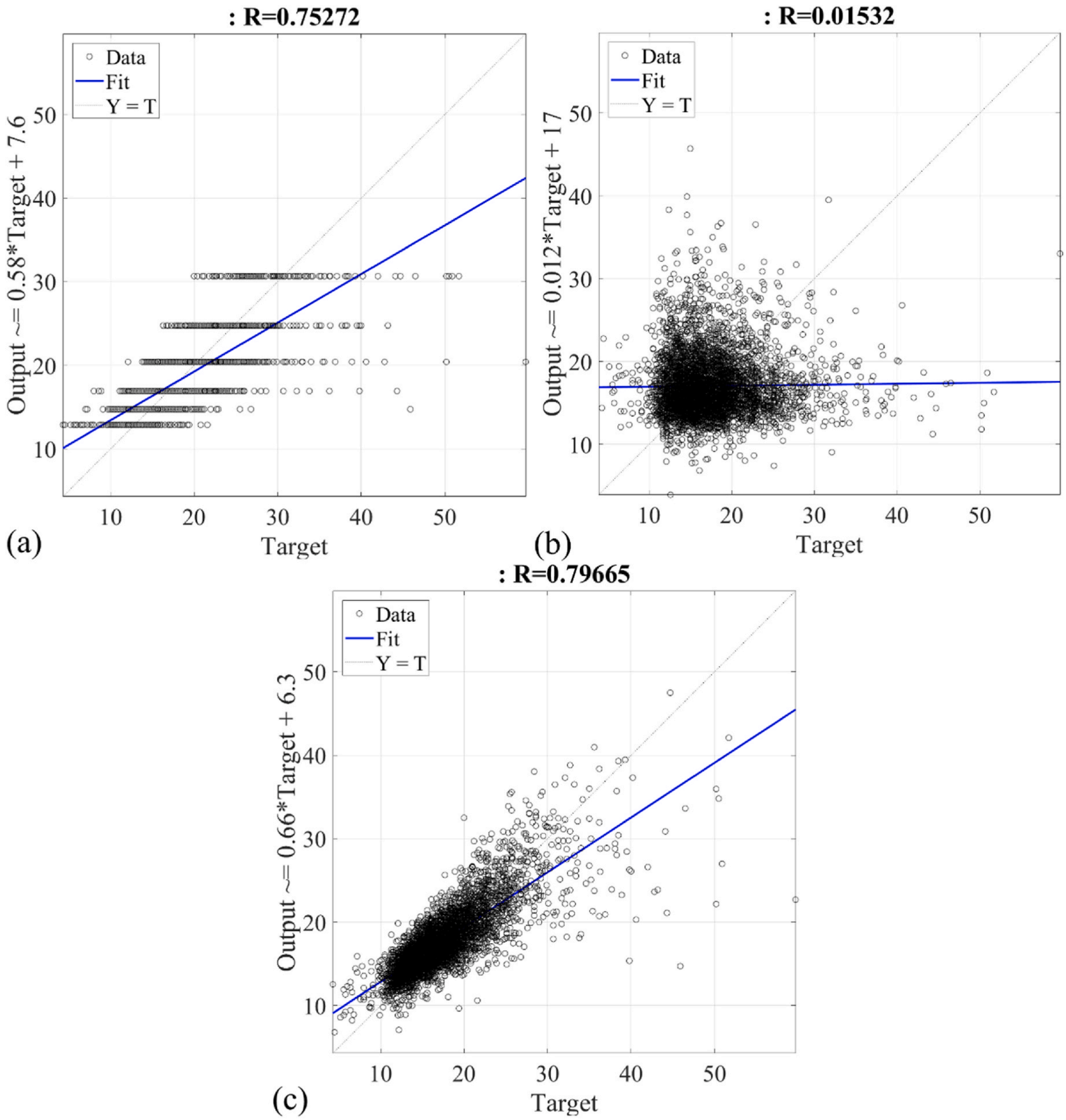

Fig. 5. The regression performance of the three investigated models, including (a): DT; (b): SVR; and (c): B-MLP, during both training and testing stages for the prediction of wind gusts.

The proposed model, called Bayesian-multilayer perceptron (B-MLP), combines an optimised sequencing block and an optimised MLP. A global (Bayesian) optimisation technique is applied to compute the optimal sequencing parameters and the MLP's indices. This aims to address the issues related to higher accuracy, robustness, overfitting, and sensitivity to small variations in existing studies within the domain. The proposed approach is validated and compared with two traditional machine learning models, SVR and DT, in predicting the daily wind gusts of Muscat International Airport from January 2010 to March 2023. The predictive power of the proposed model is evaluated using various indicators, including CC, MSE, RMSE, and R^2.

The proposed B-MLP model achieved the highest forecasting accuracy and lowest error among all compared models, with a remarkable 98.07 % correlation coefficient, owing to its ability to capture the complex, dynamic behaviour of wind gusts through optimised sequencing and Bayesian-tuned architecture. Additionally, to assess the model's robustness, we performed a T-test that demonstrated the model's resilience to the permutation of the days in the data sample considered. This quantifiable improvement highlights the effectiveness of integrating block sequencing with Bayesian optimisation as a novel strategy for reducing prediction errors and enhancing model accuracy in wind gust forecasting. The key findings of the research include the significant improvement in forecasting accuracy achieved through the proposed B-MLP model. The research also highlights the importance of selecting optimal hyperparameters in enhancing model performance. Specifically, applying this method to wind gust data from Muscat International Airport demonstrated superior accuracy, showcasing its potential for broader applications.

Despite its promising results, this study has some limitations. The model was trained and validated using data from a single coastal station, which may limit its generalisability to other geographical regions or inland environments. Additionally, the current framework uses univariate time-series input, which does not account for the influence of other meteorological variables. Future studies could extend this work by

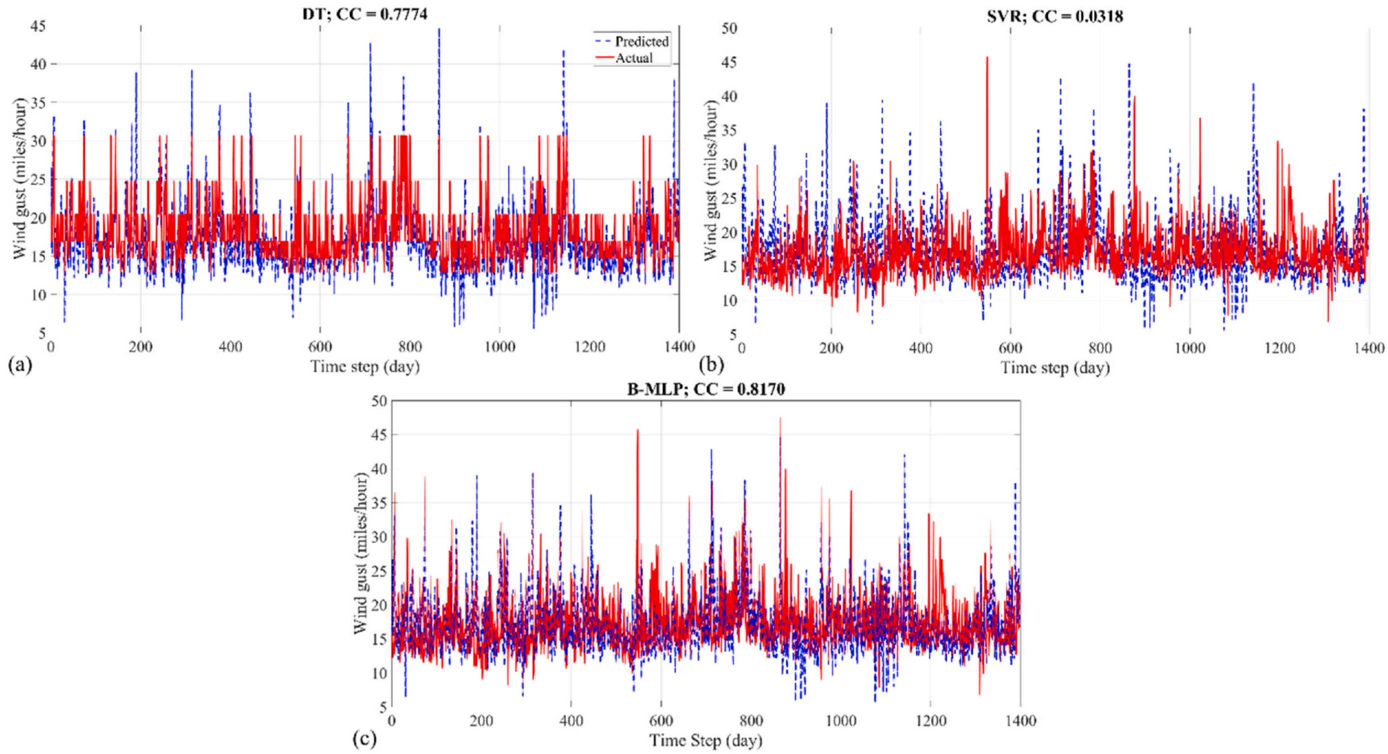

Fig. 6. The actual and predicted 1-day ahead wind gust during testing using (a): DT; (b): SVR; and (c): B-MLP.

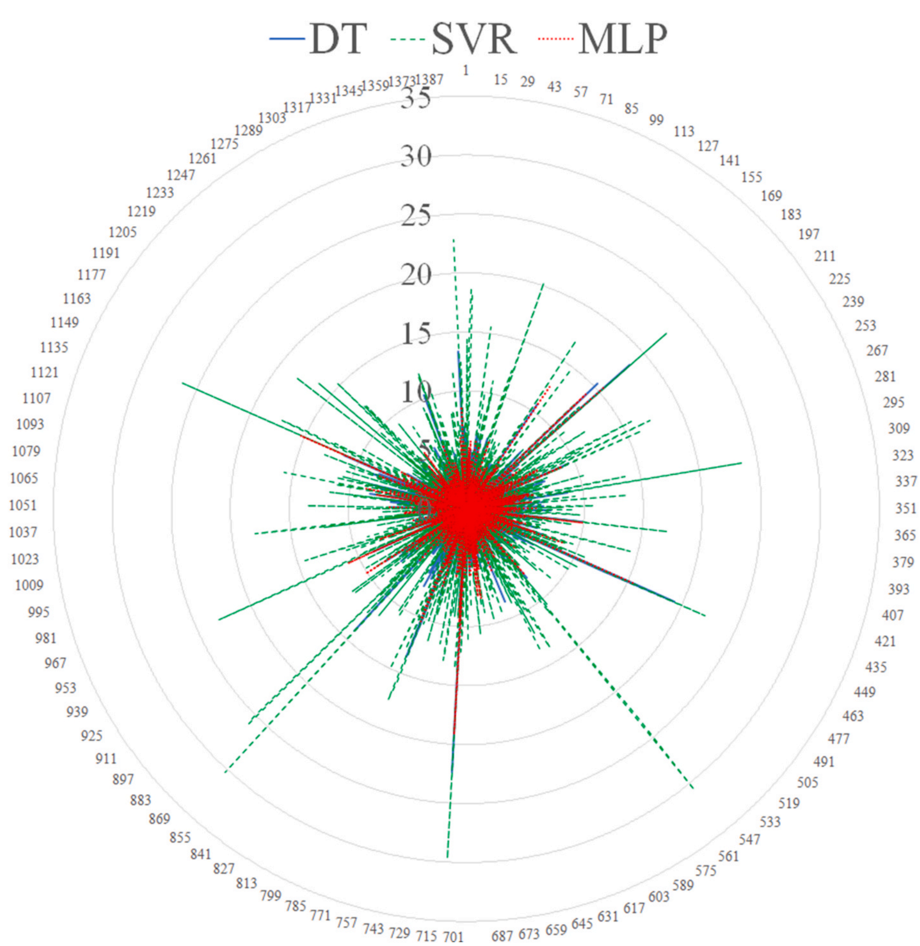

Fig. 7. The error between the actual and predicted 1-day ahead wind gusts using DT, SVR, and B-MLP.

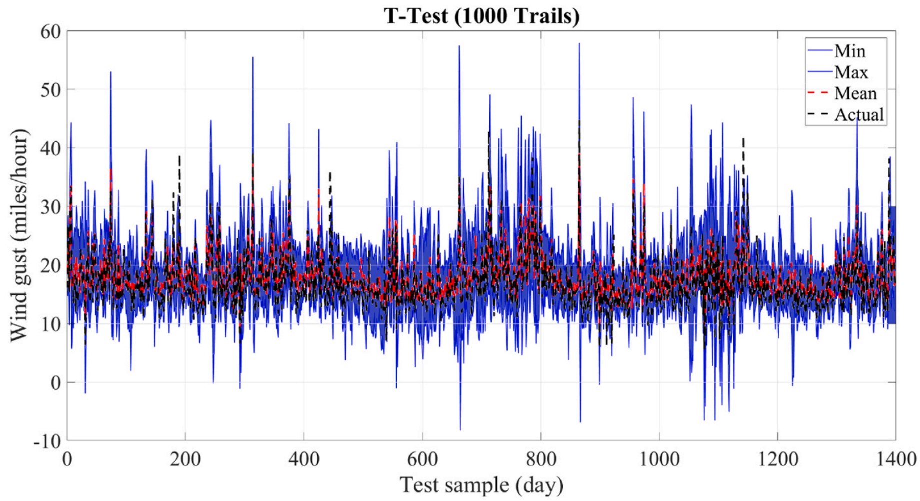

Fig. 8. The T-test results after 1000 trials, including the minimum, maximum, and mean, using B-MLP and actual wind gusts during the testing process.

Table 4
The results of the three investigated methods (SVR, DT, and B-MLP) are presented.

Model	Forecast Horizon	MAE (m/s)	CC	Key Features	Reference
LSTM-CNN	1 min ahead	0.487	–	Deep learning, convolutional layers	Xu and Wei (2022)
Attention-LSTM	6–12 h ahead	~0.6	–	Sequence attention, hybrid DL	Xing et al. (2024)
GRU + Hybrid Features	24 h ahead	~0.9	~0.79	Adaptive input encoding	Literature
B-MLP (Proposed)	1 day ahead	0.927	0.817	Block sequencing + Bayesian Optimisation	This study

Note: Conversion used — 1 mile/hour ≈ 0.447 m/s.

incorporating multivariate inputs, applying the model to multiple locations, and integrating physical atmospheric models to enhance robustness and predictive performance.

CRediT authorship contribution statement

Mohammad Reza Chalak Qazani: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Project administration, Methodology, Investigation, Formal analysis, Conceptualization. **Mahmood Al-Bahri:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Data curation. **Muhammad Zakarya:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Data curation. **Falah Y.H. Ahmed:** Writing – review & editing, Supervision, Project administration. **Amirhossein Mohajerzadeh:** Writing – review & editing, Supervision, Project administration. **Saeid Hosseini:** Writing – review & editing, Supervision, Project administration. **Mehdi Moayyedian:** Writing – review & editing, Supervision, Project administration. **Zoran Najdovski:** Writing – review & editing, Supervision, Project administration. **Houshyar Asadi:** Writing – review & editing, Supervision, Project administration.

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.

Acknowledgment

The authors gratefully acknowledge the financial support provided by Sohar University through the Internal Research Grant Program under Grant No. SUSF-2025-02.

Data availability

Data will be made available on request.

References

Attari, M.J., et al., 2024. A 45-year updating wind and wave hindcast over the Oman Sea and the Arabian Sea. Reg. Stud. Mar. Sci. 80, 103882.
Beljaars, A., 1987. The Measurement of Gustiness at Routine Wind Stations: a Review.
Bhasuru, A.S., Nagababu, G., Kachhwaha, S.S., Puppala, H., 2022. Climate change impacts the future offshore wind energy resources in India: evidence drawn from CORDEX-SA regional climate models. Reg. Stud. Mar. Sci. 56, 102717.
Boser, B.E., Guyon, I.M., Vapnik, V.N., 1992. A training algorithm for optimal margin classifiers. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory, pp. 144–152.
Brabec, M., Craciun, A., Dumitrescu, A., 2021. Hybrid numerical models for wind speed forecasting. J. Atmos. Sol. Terr. Phys. 220, 105669.
Cai, H., Jia, X., Feng, J., Li, W., Hsu, Y.-M., Lee, J., 2020. Gaussian process regression for numerical wind speed prediction enhancement. Renew. Energy 146, 2112–2123.
Cutajar, K., et al., 2023. Wind and shipping influences on sea currents around an inshore fish farm in a heavily contested mediterranean embayment. Reg. Stud. Mar. Sci. 62, 102855.
Di, Z., et al., 2019. Improving WRF model turbine-height wind-speed forecasting using a surrogate-based automatic optimization method. Atmos. Res. 226, 1–16.
Fry, C., et al., 2007. Real-time solar wind forecasting: capabilities and challenges. J. Atmos. Sol. Terr. Phys. 69 (1–2), 109–115.
Gupta, D., Natarajan, N., Berlin, M., 2022. Short-term wind speed prediction using hybrid machine learning techniques. Environ. Sci. Pollut. Control Ser. 29 (34), 50909–50927.
Haripriya, R., Misra, A., Jeyaraj, S., Ramakrishnan, B., 2021. Wave energy assessment for 39 years along the exclusive economic zone (EEZ) of India. Reg. Stud. Mar. Sci. 45, 101809.
Hastie, T., Tibshirani, R., Friedman, J., 2009. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer Science & Business Media.
Iqra, N.A., Li, J., Wang, X.H., Yang, G., 2025. Wave prediction using graph neural network at Darwin Harbour, Australia. Reg. Stud. Mar. Sci. 84, 104088.
Jiang, Z., Che, J., Li, N., Tan, Q., 2023. Deterministic and probabilistic multi-time-scale forecasting of wind speed based on secondary decomposition, DFIGR and a hybrid deep learning method. Expert Syst. Appl. 234, 121051.
Memarzadeh, G., Keynia, F., 2020. A new short-term wind speed forecasting method based on fine-tuned LSTM neural network and optimal input sets. Energy Convers. Manag. 213, 112824.
Minola, L., et al., 2020. Near-surface mean and gust wind speeds in ERA5 across Sweden: towards an improved gust parametrization. Clim. Dyn. 55 (3–4), 887–907.

Myles, A.J., Feudale, R.N., Liu, Y., Woody, N.A., Brown, S.D., 2004. An introduction to decision tree modeling. J. Chemometr.: J. Chemom. Soc. 18 (6), 275–285.
Nehama, F.P., Reason, C.J., 2021. The wind-driven response of the Zambezi River plume along the Sofala bank: a numerical model study. Reg. Stud. Mar. Sci. 46, 101891.
Pan, L., et al., 2021. Seasonal variation of the surface wind forecast performance of the high-resolution WRF-RTFDDA system over China. Atmos. Res. 259, 105673.
Pelikan, M., Goldberg, D.E., Cantú-Paz, E., 1999. BOA: the Bayesian optimization algorithm. Proc. Gene. Evol. Comput. Conf. GECCO-99 1, 525–532. Citeseer.
Perner, P., Zscherpel, U., Jacobsen, C., 2001. A comparison between neural networks and decision trees based on data from industrial radiographic testing. Pattern Recognit. Lett. 22 (1), 47–54.
Prieto-Herráez, D., et al., 2021. Local wind speed forecasting based on WRF-HDWind coupling. Atmos. Res. 248, 105219.
Rosenblatt, F., 1957. The Perceptron, a Perceiving and Recognizing Automaton Project Para. Cornell Aeronautical Laboratory.
Shang, Z., Chen, Y., Chen, Y., Guo, Z., Yang, Y., 2023. Decomposition-based wind speed forecasting model using causal convolutional network and attention mechanism. Expert Syst. Appl. 223, 119878.
Shi, J., Guo, J., Zheng, S., 2012. Evaluation of hybrid forecasting approaches for wind speed and power generation time series. Renew. Sustain. Energy Rev. 16 (5), 3471–3480.
Suman, O.P., Kumar, M., 2023. Machine learning based theoretical and experimental analysis of ddos attacks in cloud computing. In: 2023 International Conference on Device Intelligence, Computing and Communication Technologies,(DICCT). IEEE, pp. 526–531.
Tateo, A., Miglietta, M., Fedele, F., Menegotto, M., Pollice, A., Bellotti, R., 2019. A statistical method based on the ensemble probability density function for the prediction of "Wind Days". Atmos. Res. 216, 106–116.
Timofeev, R., 2004. Classification and Regression Trees (CART) Theory and Applications. Humboldt University, Berlin, pp. 1–40.
van Diepen, M., Franses, P.H., 2006. Evaluating chi-squared automatic interaction detection. Inf. Syst. 31 (8), 814–831.
Xing, W., Qi-liang, W., Gui-rong, T., Dai-li, Q., Ke, Z., 2024. A forecast model of short-term wind speed based on the attention mechanism and long short-term memory. Multimed. Tool. Appl. 83 (15), 45603–45623.
Xu, X., Wei, Y., 2022. An ultra-short-term wind speed prediction model using LSTM and CNN. Multimed. Tool. Appl. 81 (8), 10819–10837.
Zouaidia, K., Ghanemi, S., Rais, M.S., Bougueroua, L., Katarzyna, W.-W., 2021. Hybrid intelligent framework for one-day ahead wind speed forecasting. Neural Comput. Appl. 33 (23), 16591–16608.