



Research article

Probabilistic prediction intervals of short-term wind speed using selected features and time shift dependent machine learning models

Rami Al-Hajj^{1,*}, Gholamreza Oskrochi¹, Mohamad M. Fouad² and Ali Assi³

¹ College of Engineering and Technology, American University of the Middle East, Kuwait

² Faculty of Engineering, Mansoura University, Egypt

³ Independent Researcher, SMIEEE-Renewable Energy, Quebec, Canada

* **Correspondence:** Email: rami.alhajj@aum.edu.kw.

Abstract: Forecasting wind speed plays an increasingly essential role in the wind energy industry. However, wind speed is uncertain with high changeability and dependency on weather conditions. Variability of wind energy is directly influenced by the fluctuation and unpredictability of wind speed. Traditional wind speed prediction methods provide deterministic forecasting that fails to estimate the uncertainties associated with wind speed predictions. Modeling those uncertainties is important to provide reliable information when the uncertainty level increases. Models for estimating prediction intervals of wind speed do not differentiate between daytime and nighttime shifts, which can affect the performance of probabilistic wind speed forecasting. In this paper, we introduce a prediction framework for deterministic and probabilistic short-term wind speed forecasting. The designed framework incorporates independent machine learning (ML) models to estimate point and interval prediction of wind speed during the daytime and nighttime shifts, respectively. First, feature selection techniques were applied to maintain the most relevant parameters in the datasets of daytime and nighttime shifts, respectively. Second, support vector regressors (SVRs) were used to predict the wind speed 10 minutes ahead. After that, we incorporated the non-parametric kernel density estimation (KDE) method to statistically synthesize the wind speed prediction errors and estimate the prediction intervals (PI) with several confidence levels. The simulation results validated the effectiveness of our framework and demonstrated that it can generate prediction intervals that are satisfactory in all evaluation criteria. This verifies the validity and feasibility of the hypothesis of separating the daytime and nighttime data sets for these types of predictions.

Keywords: wind speed forecasting; wind energy; features selection; support vector regressors; prediction intervals; probabilistic energy forecasting; kernel density estimator.

1. Introduction

Nowadays, wind energies are sustainable and have particular potential in smart grid stability at seasonal periods of the year [1–3]. However, wind power (WP) strength forecasting is a challenge for researchers when planning to integrate WP resources into modern energy systems. This is due to non-linear and non-stationary features of the wind speed (WS) time series data [4–7]. WS forecasting is the first step of WP prediction. Therefore, reliable forecasting of WS provides adequate knowledge to draw proper decisions about the intensity of WP in a specific region.

Many WS prediction methods have been introduced to provide point predictions, also named deterministic predictions, which fail to accurately show the uncertainty and instability of WS, thus compromising the reliability of forecasted results [8–10]. Practically, deterministic forecasting systems provide only point forecasting errors without showing the probability of their correct forecasting. Therefore, a solution to quantify the uncertainty factor, such as prediction intervals (PIs), was considered an adequate need. PIs provide a range within which the observed WS is likely to fall based on a particular confidence level that reflects the model's reliability [11].

Interval prediction estimates the range of possible change in the future of a value at some point when the original data shows irregular variations. Practically, interval prediction provides upper and lower bounds of predictions at specified confidence levels. That offers decision-makers adequate uncertainty information for more accurate decision-making [12–14]. In WP applications, PI estimation is particularly beneficial for wind farm operation and maintenance engineers to plan their activities and formulate reasonable scheduling policies [15]. With the development of artificial intelligence, machine learning (ML) models became distinguished methods for both point and interval WS predictions [9,10,16]. In recent years, several ML models and architectures have been employed in WS PI estimation systems. Most of the recently proposed models are either ensemble models [17–23], hybrid models [24–27], or deep learning (DL) models [28,29]. Ensemble models combine parallel predictors of type statistical regressors and/or ML models. Hybrid models consist of combining ML models with optimization methods.

The authors in [12] proposed a hybrid model that combines least support vector machines (LSSVM) with multi-objective ant lion optimization (MOALO) algorithm to construct hourly prediction intervals of wind speed in Shandong province, China. The authors in [16] proposed a WP deterministic and interval prediction framework comprising five single ML predictors of type long-short term memory (LSTM), support vector machine (SVM), deep belief network (DBN), extreme learning machine (ELM), and convolutional neural network (CNN). A critical weight method is applied to combine the point forecasting results of the individual predictors. Then, the nonparametric kernel density estimation (KDE) method is applied to estimate the PIs around each point prediction under different confidence levels. A new deterministic and probabilistic WS forecasting framework based on explainable neural networks (NNs) is presented by Huang et al. [14]. The uncertainties in WS are statistically synthesized via the KDE method to provide the PI around deterministic forecasts [14]. The authors in [17] proposed an ensemble module with mixed frequency modeling for WS point and interval forecasting. The introduced approach applied a multi-objective optimizer to enhance the

performance of several forecasting ELM models. The authors in [18] combined ensembles of NNs and SVRs using a multi-objective optimization approach. The proposed model was examined on five WS datasets to find interval predictions. The authors in [19] applied a fuzzy information granulation technique to reduce the dimension of WS data. The proposed model uses the multi-objective dragonfly algorithm to combine four sub-models, including NNs and statistical models. The researchers in [20] employed different ML and statistical prediction methods according to modal characteristics. The authors introduced several optimization algorithms designed to enhance nonlinear prediction capabilities. Additionally, they explored a set of interval prediction schemes based on Monte Carlo theory. A quantile regression bi-directional LSTM network within an ensemble probabilistic forecasting strategy is proposed in [21] for estimating uncertainty in WS. The research in [24] applied two gated recurrent units (GRUs) to build WS PIs using prediction errors. The authors used variational mode decomposition to decompose the complex WS time series into simplified modes. The prediction error for each mode is given a weight. The prediction errors are accumulated to obtain the width of the final PIs. The optimal weights of the prediction errors are found using the particle swarm optimization algorithm. The authors in [22] proposed a clustering-based short-term WS interval prediction with multi-objective ensemble learning. A variational mode decomposition is employed to acquire the sub-sequence matrix of WS. A multi-objective optimization method is then used to choose and train an optimal model for each sub-sequence using its long-term correlation. The authors applied their proposed model for wind speed data from the national renewable energy laboratory (NREL). The authors in [23] proposed a method that combines a multi-objective artificial hummingbird algorithm, prediction interval forecasting, and statistical and gated recurrent forecasting methods.

In [25], the authors proposed a hybrid model that uses an auto-encoder-based feature extractor and bidirectional LSTM (bi-LSTM) models for short-term WS interval predictions. The reported simulations showed that feature extraction through auto-encoder is advantageous to produce narrow PIs with high PI coverage. The work in [26] combines the optimized Radial Basis Function model, the Fourier distribution, and the fast correlation based filter (FCBF) algorithm to build a WS interval prediction model. The authors applied the FCBF algorithm to filter the factors that affect the wind change, and then they introduced an improved version of the particle swarm optimization (PSO) for optimizing the RBF model. The Fourier function was used to fit the error probability distribution that is used to estimate the WS PIs. The authors in [27] proposed a forecasting model that combines a modified multi-objective tunicate algorithm MMOTA, a set of statistical and ML models, and a quantile regression QR tool for deterministic and probabilistic interval forecasts of WS. The system presented in [28] includes an empirical mode decomposition to extract the linear component of the initial WS series. Then, an autoregressive integrated moving average model (ARIMA) and back propagation neural networks BPNN are applied to produce the deterministic prediction points. Finally, an improved first order Markov chain (IFOMC) model is provided to make the uncertainty analysis of WS and produce the PIs. The authors in [30] proposed a PI model that consists of temporal convolutional networks TCN to forecast WS. The input, hidden, and output layers of the proposed model consist of a TCN layer, multiple fully connected layers using *tanh* activation function, and an end-to-end sorting layer. Table 1 summarizes the above-mentioned models.

Table 1. A summarized overview of the surveyed PI forecasting methods.

Models	Structure	Ref.	Dataset	Advantages	Main Results
LSSVM	Hybrid + optimization methods	[12]	Hourly wind speed data from Shandong province, China.	Simultaneous construction of lower and upper bounds.	Seasonal models with good coverage and interval width.
ARIMA + NNs	Hybrid ML models	[13]	10-min average WS, China	Operative method for ultra-short term WS PI forecasting.	The higher PINC value leads to the good coverage probabilities and wide bandwidth.
Explainable NNs	Parallel ensemble	[14]	1-hour ahead weather and WS data	The model expresses how the inputs affect the outputs by mathematical modeling	The ACE of the introduced model provides the smallest deviations to the nominal confidence levels, mainly within high confidence levels.
LSTM + ELM + SVM + DBN + CNN	Ensemble + optimized combination	[16]	15-min WS data.	The framework is robust against variations since it is an ensemble approach	The overall ensemble performance is better than each single model with various enhancements.
ELM-based models	Ensemble + optimized combination	[17]	10-min WS. Chengde and Penglai-Shandong, China	Novel modeling technique, advanced ensemble model with mixed frequency modeling.	PI estimation using different optimization objectives.
SVM + ELM + MLP	Ensemble + optimized combination	[18]	10-min WS data from four cites in Dalian, China.	Decomposition and reconstruction technology is applied to remove the high-frequency noise.	The combined PI estimations overperformed all single models for one step and multi steps predictions.
NNs + ELM + ARIMA.	Ensemble + optimized combination	[19]	10-min wind speed. Penglai-Shandong, China	Preprocessing steps transform the high-dimensional WS data into low-dimensional subsets using fuzzy information granulation	Improvement in MAPE for all multi-steps forecasting against single models.
NNS + Markov Chain Monte Carlo Method	Parallel ensemble	[20]	5-min Changma dataset (China), 10-min Sotavento dataset (Spain).	Back propagation neural network is optimized by improved cuckoo algorithm, strong global optimization.	Successfully identify several modal characteristics and enhance prediction accuracy. The Monte Carlo method is applied to estimate PIs, Strong prediction applicability

Continued on next page

Models	Structure	Ref.	Dataset	Advantages	Main Results
Bi-directional (LSTM)	Ensemble probabilistic forecasting	[21]	10-min wind speed. Penglai-Shandong, China.	Reliable uncertainty forecasts, good global optimization ability, and good Excellent fitting ability.	The PI coverage probability provided by the system is above 97%, and the correctness is enhanced by 24.21% against single models
GRU ensemble	Ensemble + optimized combination	[24]	30-min wind speed records from Boston and Huston wind fields	Strong applicability, Strong stability of the result	The constructed PIs are of high credibility with PICP indices higher than PINC (90%). The performance is sensitive to changes in locations and seasons.
GRU + SVR + ARIMA	Clustering-based ensemble learning	[22]	Two 10-min WS data from NREL (California, and Washington)	Effective optimization for ensemble learning, clear forecasting process.	Robust prediction on onshore and offshore scenarios. More than 4.77% improvement on PICP.
LSTM + GRU + ARIMA	Ensemble + optimized combination	[23]	10-min WS. Penglai-Shandong, China	Capturing linear and nonlinear features of time series data.	Heterogeneous interval prediction methods are combined by multi-objective artificial hummingbird algorithm.
Bi-LSTM	Hybrid BLSTM + auto-encoder	[25]	Two 30-min WS data from NREL (Lake Huron, and Pennsylvania)	Feature extraction through auto-encoder is effective, Good stability.	Feature extraction through auto-encoder, high PI coverage with narrow intervals.
FCBF + RBF	Hybrid ML models	[26]	5-min Changma dataset of wind speed	Performant FCBF based feature selection, the solution is effective and feasible.	Hybrid ML with statistical knowledge, the average width of the PI is less than 3 m/s.
MMOTA + QR + ANN	Hybrid + optimization methods	[27]	5-min Changma dataset (china), 10-min Sotavento dataset (Spain).	The filter of high-frequency noise improve the effectiveness,	The empirical research demonstrates the optimal forecasting of WS.
ARIMA + BPNN+IFOMC	Hybrid ML models	[28]	10-min average WS from China	higher accuracy and efficiency, Strong stability of the result.	Hybrid solution with efficient computation and convergence.
TCN	Deep Learning	[30]	Two 15-min WS data from NREL (Patterson, and San Francisco).	Good stability and reliability, optimal temporal models.	PIs with a satisfactory coverage probability, satisfactory PIPC for both benchmarking datasets.
LSTM	Deep Learning	[29]	Four 10-min WS data from NREL (Maine, Rhode Island, North Carolina, Virginia)	New loss functions for gradient descent back propagation, good algorithm convergence	Deep learning PI forecasting with good fitting ability, comparable advantageous convergence time.

The above mentioned models do not differentiate between daytime and nighttime data records. Our hypothesis in this study assumes that the WS and direction are influenced by weather parameters such as air pressure, air temperature, air density, and many others. The values of those parameters related to daytime shifts are quite different from those related to nighttime ones. In this work, we introduce and train independent daytime and nighttime models for deterministic predictions. Then we find prediction intervals of WS in short-term horizons. To the best of our knowledge, considering daytime and nighttime records separately for designing ML models to estimate PIs of WS has not been explored and lacks comprehensive investigation. Thus, we aim to address this problem and introduce a new probabilistic interval prediction model for short-term WS forecasting. The proposed model investigates a hybrid intelligent approach based on independent short-term predictors of type SVRs, feature selection module, and KDE technique to estimate PIs of WS.

In the first step, we arrange the records of weather parameters into daytime and nighttime records. The automatic selection of the most relevant features for each of the day and night-related models is implemented in a pre-processing phase. Feature selection techniques such as recursive features elimination and univariate selection have been tested and compared. Then, the sliding window method is applied to convert the time-series prediction task to a regression task. All the prediction models are trained for 10-minutes-ahead forecasting of WS.

The main contribution of this work is as follows:

(1) A new prediction approach that consists of independent models for daytime and nighttime shifts is proposed to estimate PIs for short-term forecasting of WS.

(2) The relevant features for each time shift are automatically selected using two different techniques, namely the Univariate Selection US and the recursive feature elimination (RFE). The deterministic prediction models investigate the stochastic relationship among the input features related to each shift and the estimated WS 10-minutes ahead.

(3) The KDE method is applied for each time shift separately to analyze the prediction errors of the related point prediction models without setting a hypothesis about the distribution of the WS prediction errors in advance.

(4) The simulation results indicate that considering independent daytime and nighttime regression models and applying suitable feature selection provides better PI forecasting than using global prediction models.

The proposed framework was validated on testing data records with short time horizons. The obtained scores of simulation works demonstrated the effectiveness of the introduced framework. Three evaluation metrics, specifically the prediction interval coverage probability (PICP), the Prediction interval normalized average width (PINAW), and the coverage width-based criterion (CWC), have been designated to evaluate the proposed WS prediction interval framework.

The remaining sections of this work are organized as follows: In Section 2, we introduce the major algorithms and methodology that have been applied. In Section 3, we detail the structure of the proposed approach. In Section 4, we present and discuss the simulations and results. Finally, in Section 5, we present the conclusions and perspectives of this work.

2. Related methodology

2.1. Data cleaning: quartile method to remove outliers

The quartile method (QM) is a statistical technique that identifies and eliminates data points that are different from the majority of a numerical dataset. This method involves four major steps:

Step 1: The QM arranges the data values of a variable x_i in ascending order $X = \{x_1, x_2, \dots, x_n\}$. Then the Quartile points Q_1 , Q_2 , and Q_3 , which divide the ordered data into four quartiles, are calculated as follows:

$$Q_2 \text{ (the median)} = \begin{cases} \frac{x_{\frac{n+1}{2}}}{2} & \text{if } n \text{ is odd} \\ \frac{x_{\frac{n}{2}} + x_{\frac{n+2}{2}}}{2} & \text{if } n \text{ is even} \end{cases} \quad (1)$$

The second quartile Q_2 divides the ordered dataset into two parts D_1 and D_2 , Where the first Quartile Q_1 and the third Quartile Q_3 are the medians of D_1 and D_2 respectively.

Step2: The Inter Quartile Range (IQR) of the dataset X , which is the difference between Q_1 and Q_3 is computed as follows:

$$IQR = Q_3 - Q_1 \quad (2)$$

Step 3: The lower-bound B_l and the upper-bound B_u are found to identify outliers:

$$B_l = Q_1 - 1.5 IQR \quad (3)$$

$$B_u = Q_3 + 1.5 IQR \quad (4)$$

All data points outside the interval $[B_l, B_u]$ are suggested as outliers in the dataset X . Figure 1 is a boxplot that illustrates the quartile points and the IQR interval computed by the QM.

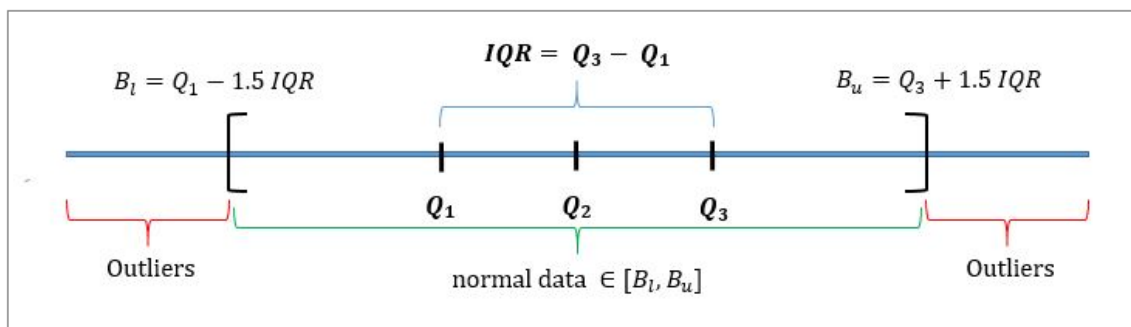


Figure 1. A typical Boxplot and the calculated IQR using the QM method.

2.2. Feature selection

Feature selection involves the automatic choosing of the most relevant features of a set of features. Features selection helps in reducing the space dimensionality of datasets and enhances the overall performance of the ML algorithms. In addition, it makes it easier to understand the underlying relationships between the selected features and the target variable [31,32]. The feature selection techniques that are applied to prepare input data for the ML models are:

(1) Univariate selection: This is an effective method for feature selection that incorporate statistical tests to select the features that have a resilient relationship with the output [31]. It evaluates each feature individually to determine its relevance for predicting the target variable. The first step of the univariate selection process evaluates the relevance score of each feature using a statistical test to measure its correlation with the target variable. Pearson's correlation and F-Test (ANOVA) are among the adopted features scoring methods for regression tasks. Based on the scoring, and then the ranking of the features, the top k ones are selected. Alternatively, features can be selected based on a specific significance level ($p\text{-value} < 0.05$). The univariate selection method is computationally efficient, especially for high-dimensional datasets.

(2) RFE: This method removes the features that do not have a strong relation with the output in a recursive way [31,32]. It aims to find a subset of features that results in the best performance for the model. This wrapper method uses the accuracy of an ML model to detect the features or the subset of features that contribute the most in predicting the output variable. The RFE algorithm works as follows:

- 1) Train an ML model on the initial feature set that includes all features.
- 2) Evaluate the importance of each feature based on a specific criterion.
- 3) Rank the current set of features with respect to their importance scores computed in Step (2)
- 4) Remove the least important feature(s) from the current set of features
- 5) Repeat Steps 1–4 with current features until a stop criterion is reached (e.g., a predefined number of features is reached, or the performance of the wrapper model stops improving).

The evaluation of feature importance is a crucial step in the RFE method, and it varies based on the adopted ML model. For the SVRs, there are a few techniques that are reported in the literature as estimators of feature importance. These include the Permutation Importance that can be applied with both linear and non-linear SVRs. This technique randomly permutes the values of a feature and evaluates the changes in the model's performance score for instance, the mean squared error. The features that, when permuted, cause a larger decrease in performance are considered important ones. Another approach to evaluate feature importance is the weight vector approach. This approach is applicable to linear SVRs where the feature importance can be estimated from the weight vector (coefficients) learned by the model.

We run all the experimental simulations using the Scikit-learn 0.24.2 framework that requires Python 3.6 or successor versions [33].

2.3. Support vector regressors

SVRs models are ML tools to address regression problems. Currently, they are competitor methods in approximating continuous functions [34–36]. They are a type of SVM designed to model regression tasks.

The objective of SVRs is to find a flexible tube around the function to be approximated for as many data points as possible within a specified margin of tolerance defined by a parameter epsilon (ϵ) [34]. In SVR models, support vectors are the key data points that lie on or outside the boundaries of the epsilon tube defined by the parameter epsilon ϵ around the predicted function, as shown in Figure 2. Those points are the only ones that determine and affect the final position and orientation of the regression function. This makes the model less sensitive to outliers [35].

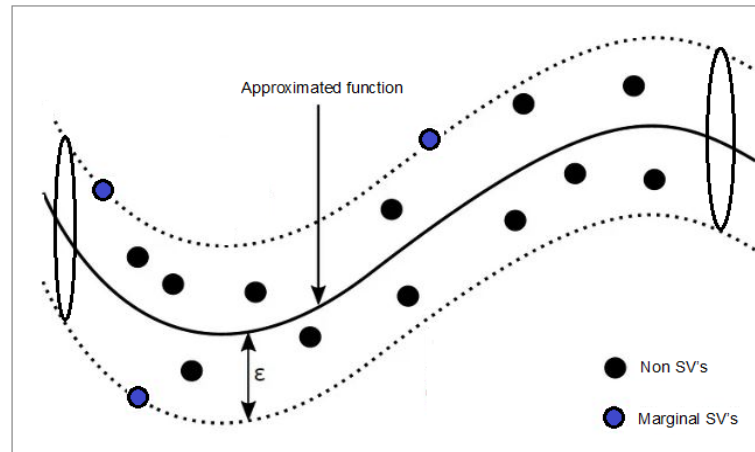


Figure 2. Support vector regression (SVR) with epsilon parameter and support vector points.

SVR models also incorporate kernel functions that allow them to handle non-linear relationships among variables. Commonly used kernels include polynomial kernel, radial basis function (RBF) kernel, and sigmoid kernel [34]. These kernels map the input features' space into a higher-dimensional space, enabling the SVR to capture complex patterns and associations among data records.

In addition to their robustness to outliers and ability to model non-linear relationships between the input variables and the target output, the SVRs are advantageous thanks to the independence of their computational complexity with regards to the dimension of vectors of input data.

2.4. Kernel density estimation (KDE)

The KDE is a non-parametric estimation method that is commonly used for data fitting when the potential probability density function (PDF) cannot be found [37,38]. Statistically, the PDF of a random variable describes the likelihood of that random variable taking a specific value. The KDE consists of placing a soft kernel function at each data point. It then sums these kernels to create a smooth estimate of the underlying PDF of the data points' distribution. This statistical technique does not assume any particular distribution of the data. Our study uses the KDE method to create a random distribution model of the WS forecasting errors. Practically, we estimated the PIs of the WS by analyzing the distributions of forecasted WS errors produced by the ML prediction models.

In general, the PDF is estimated as follows:

Suppose $S = \{s_1, s_2, s_3, \dots, s_n\}$ is a set of n sample points of WS prediction errors. Then, the PDF of the WS errors can be written as follows:

$$\hat{f}(s, h) = \frac{1}{n \cdot h} \sum_{i=1}^n k\left(\frac{s - s_i}{h}\right) \quad (5)$$

where $K(s, h)$ represents the kernel function, and h represents the bandwidth parameter that controls the smoothness of the estimation.

Common kernel functions include Gamma kernel, uniform kernel, Gaussian kernel, and others are used with the KDE method [37,38]. In our approach, we selected the Gaussian kernel as the kernel function k when we estimated the PDF of error distribution using the KDE. The $k(\cdot)$ is expressed by:

$$k(w) = \frac{1}{\sqrt{2\pi}} e^{-\frac{w^2}{2}} \quad (6)$$

Thus, the estimated PDF can be expressed by:

$$\hat{f}(s, h) = \frac{1}{\sqrt{2\pi}nh} \sum_{i=1}^n e^{-\frac{1}{2}\left(\frac{s-s_i}{h}\right)^2} \quad (7)$$

where h is the bandwidth parameter that determines the width of the distribution interval of the prediction error, and s_i is the i^{th} sample of the WS prediction error.

3. Proposed approach

In this section, we describe the general structure of the hybrid probabilistic approach for estimating the WS PIs. The overall structure of the proposed approach is depicted in Figure 3. The input of the proposed model is the climatological parameters and the WS data records at time t . The result outputs are the lower bound L_{t+1}^α and Upper bound U_{t+1}^α of the PIs at time $t + 1$ with a corresponding confidence level $100(1 - \alpha)\%$.

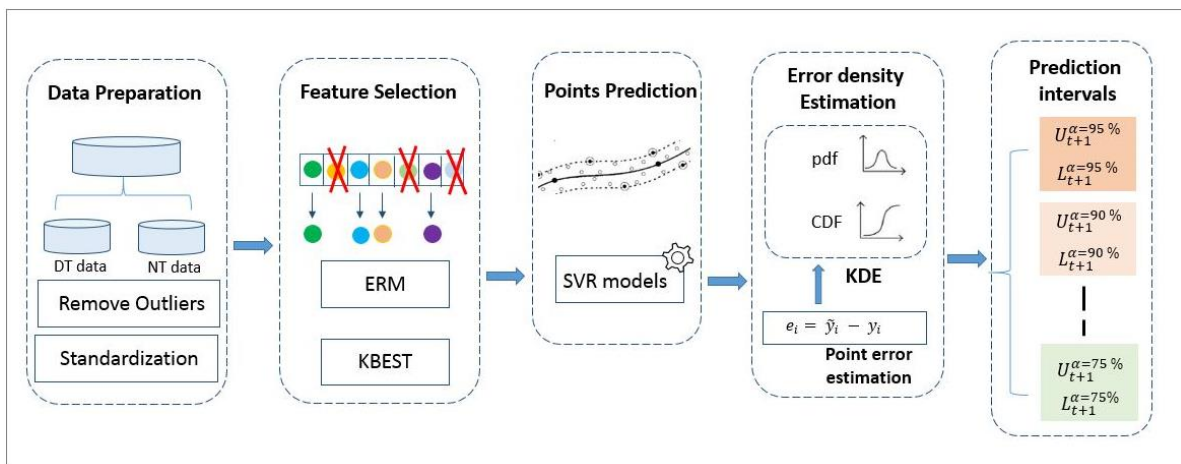


Figure 3. The general pictogram of the WS PIs forecasting model.

As illustrated in Figure 3, the general structure of the WS prediction approach introduced in this work consists of four main steps. The first is data processing and preparation, including the separation of daytime and nighttime data records. The second is feature selection to find the most relevant feature for each time shift of the day. The third is the point prediction module using SVR regressors. Finally, the fourth is the PIs forecaster module. The following sections detail each component of the proposed architecture.

3.1. Data processing module

The data pre-processing and preparation module involves three main steps as follows:

(1) Separate daytime and nighttime data records:

The dataset is divided into daytime and nighttime datasets using solar radiation amount in each data record.

(2) Remove outliers:

The outliers of each time shift dataset are removed using the QM module described in Section 2.1. The QM is applied separately to the daytime and nighttime datasets.

(3) Data scaling and normalization:

The Data normalization technique aims to rescale features and variables to a common range, typically between 0 and 1. Practically, it is computed by subtracting the minimum value from each data point and then dividing by the range (*maximum* – *minimum*) as shown in Eq (8). This technique is commonly named MinMax scaling.

Normalization is useful when dealing with different scales or units of measurement across features since it ensures that all features have equal contribution to the analysis or mode.

$$p_{scaled} = \frac{p - \min(P)}{\max(P) - \min(P)} \quad (8)$$

where p is the computed value of parameter P , and $\min(P)$ and $\max(P)$ are the lowest and highest values of the parameter P .

(4) Sliding window:

This technique permits to convert a prediction task into a regression one. The sliding window re-arranges the time series data records by considering the weather parameters and the WS of the k^{th} record as input vector, and the WS of the $(k + 1)^{th}$ record as the related target output. Figure 4 depicts the sliding window method.

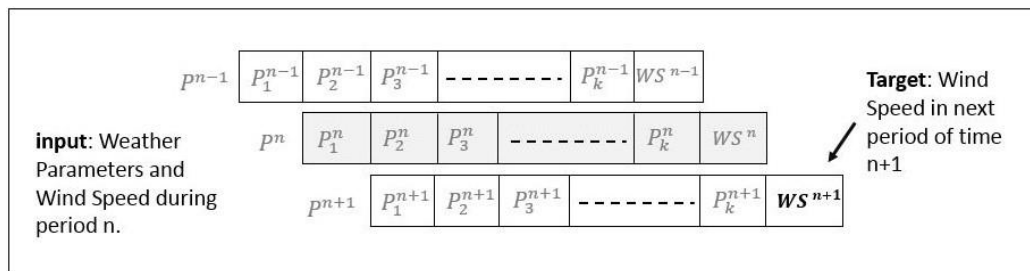


Figure 4. The sliding window method models a time-series prediction task to a regression one.

3.2. Feature selection and extraction module

This task involves identifying and selecting the most influential features (variables) with significant contribution to accurate predictions. In general, the selection of appropriate features can have a substantial impact on an ML model's performance and computational efficiency. WS forecasting models typically utilize a wide range of input weather parameters. However, not all these parameters are equally important or contribute equally to the prediction accuracy. Therefore, an effective feature selection enables the elimination of irrelevant or redundant parameters.

In the proposed framework, we apply two feature selection techniques. The univariate selection (US) is a filter method based on statistical measures, whereas the RFE is a wrapper method that evaluates subsets of features recursively. We believe that the relevant features in the daytime and nighttime datasets may be different. Therefore, the two adopted methods are applied to daytime and nighttime sub-datasets separately, as illustrated in Figure 5.

As shown in Figure 5, the US method is applied to both daytime and nighttime datasets to produce two new datasets DT-US and NT-US. On the other hand, the RFE method is also applied to both

daytime and nighttime datasets to produce two new datasets DT-RFE and NT-RFE.

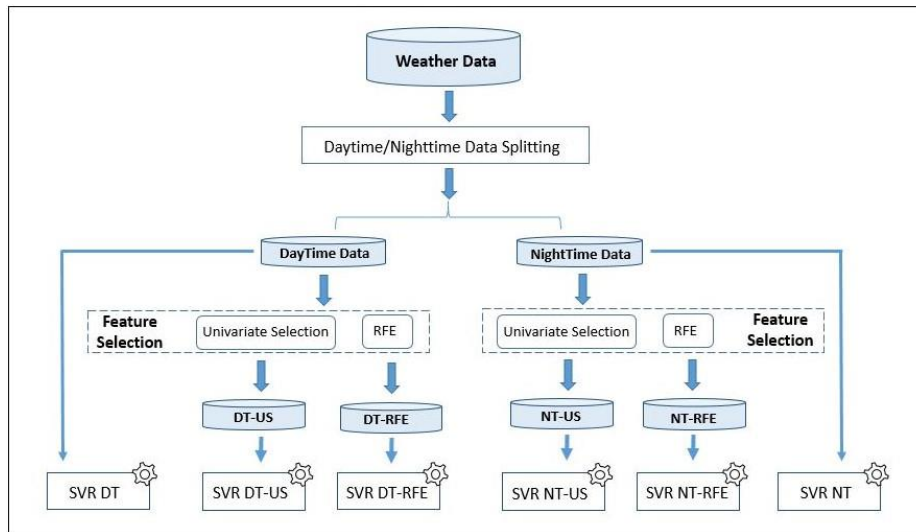


Figure 5. The US and the RFE are applied to the daytime and nighttime data sets to produce two sub-datasets for each.

3.3. Point prediction module

The point prediction module consists of SVR-based models for deterministic wind speed prediction. In our research, we trained a set of three independent SVR-based predictors for each time shift. A particular SVR predictor related to a time shift is trained on the related dataset either with all features or with selected features. Besides, we trained three reference global SVR models on global datasets that do not separate daytime and nighttime data records. All the examined SVR models have RBF kernels capable of modeling non-linear relationships between target and input variables [37,38]. The nine obtained deterministic prediction models are as follows:

3.4. Interval prediction module

Table 2. The SVR-based predictors for point predictions of WS.

model	Training dataset	Description
SVR-DT-US	DT-US	Trained on a daytime dataset with features selected by the US method.
SVR-DT-RFE	DT-RFE	Trained on a daytime dataset with features selected by RFE method
SVR-DT	DT	Trained on a daytime dataset with all features
SVR-NT-US	NT-US	Trained on a nighttime dataset with features selected by US method
SVR-NT-RFE	NT-RFE	Trained on a nighttime dataset with features selected by RFE method
SVR-NT	NT	Trained on a nighttime dataset with all features
SVR-Global-US	Global-US	Trained on a global dataset with features selected by US method
SVR-Global-RFE	Global-RFE	Trained on a global dataset with features selected by RFE method
SVR-Global	Global	Trained on a global dataset with all features

The proposed approach applies the KDE method to estimate the PDF that fits the regression errors of WS point predictions generated by the SVR models to estimate the PIs of WS. Recently, the KDE method has been considered a statistical tool to analyze the characteristics of data distribution without using prior distributions [39]. Therefore, the potential impact of the hypothesis on WS forecasting error and prediction accuracy can be suitably reduced.

After training the SVRs related to the daytime and nighttime time shifts using separate training datasets, the errors of the deterministic predictions of the WS are evaluated using validation datasets, also related to daytime and nighttime shifts. Consequently, a sequence of distinct prediction errors E_d and E_n related to the two time shifts of the day are collected, respectively:

$$E_d = \{e_d^1, e_d^2, \dots, e_d^D\}, \quad E_n = \{e_n^1, e_n^2, \dots, e_n^N\}$$

where D represents the number of validation dataset records related to the daytime shift, and N is the number of records in the validation dataset related to the nighttime shift.

The KDE method is then applied on E_d and E_n to estimate the PDF for each set of point predictions' errors respectively. Therefore, Eq (7) is applied twice as follows:

$$\hat{f}(s, h_d) = \frac{1}{\sqrt{2\pi} n h_d} \sum_{i=1}^D e^{-\frac{1}{2} \left(\frac{s - e_d^i}{h_d} \right)^2} \quad (9)$$

$$\hat{f}(s, h_n) = \frac{1}{\sqrt{2\pi} n h_n} \sum_{i=1}^N e^{-\frac{1}{2} \left(\frac{s - e_n^i}{h_n} \right)^2} \quad (10)$$

Equation (9) estimates the PDF of the prediction errors provided by the daytime SVR model using the daytime validation dataset. On the other hand, Eq (10) estimates the PDF of the prediction errors provided by the nighttime SVR model using the nighttime validation dataset.

The h parameter in Eq (7) represents the bandwidth that determines the width of the distribution interval of the prediction error. In the proposed approach, we assume that an appropriate bandwidth may differ from one set of errors to another. Therefore, we consider the possibility of having two different bandwidth values, h_d and h_n for the daytime and nighttime estimated PDF as shown in Eqs (9) and (10), respectively. In general, selecting an appropriate value of the bandwidth parameter h controls the smoothness of the estimation, and provides the balance between the variance and bias in the results. This in turn helps minimize the error of estimation [37]. In this work, we used the trial and error method to choose the most appropriate bandwidth value for each error dataset.

Using the estimated PDFs, we computed the cumulative distribution function (CDF) that represents the probability that a random variable will take a value less than or equal to a given point [37,38]. In this case, the random variable is the point prediction error. By integrating the PDF of the point prediction error distribution, the CDF accumulates the probabilities up to a specific value, allowing the estimation of the PIs.

In the developed model, we compute the CDF to obtain the fluctuation range of the prediction error given a PI nominal confidence (PINC) that is equal to $100(1 - \alpha)\%$. The fluctuation range can be expressed as follows:

$$I_\alpha = [F(\alpha), F(1 - \alpha/2)] \quad , \quad \text{where} \quad 0 < \alpha < 1 \quad (11)$$

Therefore, the estimated PI related to the point predicted target of WS with $100(1 - \alpha)\%$ confidence level is expressed as follows:

$$[\hat{L}_i^{(\alpha)}(x_i), \hat{U}_i^{(\alpha)}(x_i)] = [\hat{y}_i^{(\alpha)} - F(\alpha), \hat{y}_i^{(\alpha)} + F(1 - \alpha/2)] \quad (12)$$

where $\hat{y}_i^{(\alpha)}$ is the point (deterministic) prediction of WS related to the input x_i within the $100(1 - \alpha)\%$ confidence level.

The general chart of the suggested probabilistic prediction approach is divided into three main phases, as shown in Figure 6. In the first phase, we use the training dataset of one of the time-shift data records to train its related SVR-based deterministic prediction model. The optimal version of the trained SVR model for each time shift is saved for the next phase. In the second phase, the parameters of the PDF of the error distribution are estimated using the KDE method. The prediction errors' datasets for the daytime and nighttime shifts are obtained by examining the distinct trained models in the previous step using the validation data set related to daytime and nighttime, respectively. Consequently, the obtained error datasets and the estimated PDF functions related to each time shift are adopted to calculate the CDF of the errors related to each time shift. Subsequently, the probabilistic WS prediction models for PI estimation are statistically established. In the final phase, the probabilistic WS framework is tested using the testing datasets to verify its forecasting performance. The proposed framework provides the WS forecasting uncertainties, which are probabilistically represented as a set of quintiles, as shown in Eq (12).

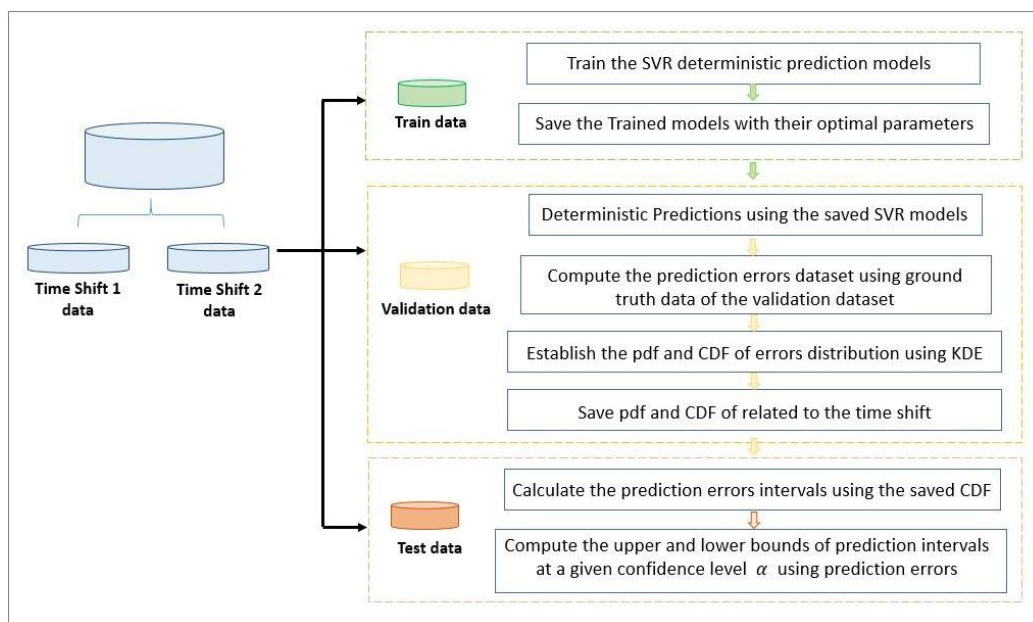


Figure 6. General flowchart of the phases of the probabilistic framework based on the SVRs.

4. Simulation and discussion

4.1. Evaluation criteria of prediction performance

4.1.1. Evaluation metrics of deterministic (point) prediction

Three statistical evaluation metrics are adopted to quantify the forecast performance of the point prediction models. These indices are defined as follows:

The root mean square error (RMSE) estimates how large the errors are between the predicted and real values by estimating the average distance of the estimated values from the real values. This index is computed by the formula:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (H_{p,i} - H_i)^2}{n}} \quad (13)$$

where $H_{p,i}$ designates a forecasted output and H_i is the measured value related to that output.

Mean square error (MAE): the MAE estimates the average magnitude of prediction errors without considering their direction. MAE is not sensitive to outliers since it does not square the residuals. This index is computed by the formula:

$$MAE = \frac{1}{n} \sum_{i=1}^n |H_{p,i} - H_i| \quad (14)$$

The mean bias error (MBE) computes the average bias in a model's predictions and indicates whether a prediction model tends to over-predict or under-predict the actual values.

This index is computed by the formula:

$$MBE = \frac{1}{n} \sum_{i=1}^n (H_{p,i} - H_i) \quad (15)$$

The mean absolute percentage error (MAPE) computes the mean absolute percentage difference between the estimated values and the exact ones. The MAPE is computed as follows:

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{H_{p,i} - H_i}{H_i} \right| \times 100 \quad (16)$$

For all the above-mentioned metrics the lower values designate a better model performance.

4.1.2. Evaluation metrics of PI-based forecasting performance

Practically, the PI-based forecasting performance can be evaluated based on the prediction width of upper and lower bounds. Three evaluation indices were selected to analyze the forecast performance of the PI-based forecasting models. These three indices are defined as follows:

The Prediction interval coverage probability (PICP) quantifies the percentage of times the actual values fall within the prediction intervals. Practically, it assesses the accuracy of the intervals in terms of coverage. The PICP is computed as follows:

$$PICP = \frac{1}{N} \sum_{i=1}^N c_i, \quad c_i = \begin{cases} 1, & p_i \in [L_i, U_i] \\ 0, & p_i \notin [L_i, U_i] \end{cases} \quad (17)$$

where N is the total number of testing samples, and c_i is a Boolean variable that is equal to 1 if the sampling point $p_i \in [L_i, U_i]$, and 0 otherwise.

An ideal PICP should align closely with the confidence level of the intervals, such as 85% for 85% confidence intervals. This alignment suggests that the PIs accurately capture the anticipated proportion of actual data. Conversely, a low PICP suggests that the PIs are too narrow, often failing to encompass the data (under-coverage). Furthermore, a high PICP implies that the PIs are too wide (over-coverage), incorporating an unnecessary amount of uncertainty.

Prediction interval normalized average width (PINAW) computes the average width of the PIs relative to the range of the data. The PINAW is computed as follows:

$$PINAW = \frac{1}{N} \sum_{i=1}^N \frac{U_i - L_i}{Y_{max} - Y_{min}} \quad (18)$$

where U_i and L_i represent the upper and lower bounds of the i^{th} estimated PI, whereas Y_{max} and Y_{min} are the maximum and minimum values of the target output in the testing dataset.

A lower PINAW score indicates narrow PIs relative to the data's range, suggesting more precise predictions. However, balancing lower PINAW with PICP is crucial to ensure that narrower intervals do not lead to under-coverage.

Coverage width criterion (CWC) is a comprehensive index that incorporates both the coverage probability PICP and the normalized width of the PIs PINAW metrics. It penalizes both the failure to cover the actual data points (low coverage) and the wide PIs simultaneously. The CWC is computed as follows:

$$CWC = PINAW(1 + \gamma e^{-\eta(PICP - \mu)}) \quad (19)$$

$$\gamma = \begin{cases} 0, & PICP \geq \mu \\ 1, & PICP < \mu \end{cases}$$

where μ is the target confidence level, e.g., 90%, and η is the penalty factor that weighs the importance of achieving the target coverage probability. Often, η is tuned empirically. Lower values of CWC are preferable as they indicate that the PIs are both narrow and achieve good coverage accuracy.

4.2. Data collection and pre-processing

We conducted all our simulation works on time series records collected from the AUMET dataset that consists of historical weather and solar radiation time series data records. The AUMET dataset is collected through the AUMET weather station at the American University of the Middle East AUM in Kuwait. Each data record of the AUMET dataset consists of the measures of sixteen climatological parameters with 5-minutes resolution. In our experiments, we considered a subset of nine parameters that we designate in Table 3.

Table 3. The examined Weather and Solar data features from the AUMET dataset.

Parameter	Unit	Description
Air_Density	Kg/m ³	Density of the air
Air_Temp	C°	Temperature of the air at three meters above the surface
Corr_Wind_Dir	deg.	corrected wind direction
Pressure	hPa	Air pressure
Relative_Wind_Dir	angle°	Relative wind direction
Relative_Wind_Speed	m/s	Wind speed
Relative_Humidity	pct	Relative humidity
Solar_Radiation	w/m ²	Amount of solar radiation
Surface_Temp	C°	Air temperature at 10 cm above the surface

The AUMET dataset used in our simulations comprises a total of 116,188 records. During the preprocessing phase, the resolution was reduced from 5 minutes to 10 minutes. The dataset was divided into two subsets: a training subset containing 22,415 records and a hold-out subset with 13,237 records for testing. Additionally, two separate training subsets for daytime and nighttime were created by filtering the solar radiation parameter based on a specific threshold. This process resulted in three training subsets: A global subset combining both daytime and nighttime records, a daytime subset, and a nighttime subset. Similarly, the testing subset was divided into separate daytime and nighttime datasets using the same solar radiation threshold applied to the training data.

An overview of the resulting subsets is provided in Table 4.

Table 4. Training, validation, and testing subsets.

Subset	Size	Utilized to
Daytime_train	12,907	Train the daytime model
Daytime_validation	5532	Validate the daytime model
Daytime_testing	6731	Test the daytime model
Nighttime_train	9508	Train the nighttime model
Nighttime_validation	4076	Validate the nighttime model
Nighttime_testing	6506	Test the nighttime model
Global_Train	22,415	Train the global model
Global_validation	9608	Validate the global model
Global_Testing	13,237	Test the global model

We used the *MinMax* scaling Eq (8) to reduce the difference in the scales of the parameters' values in the training and testing data.

We run the features selection methods to select the top five most prominent features for each dataset. In the simulation works, we explored various values for the number of features to be chosen, denoted as $k, k \in \{5, 6, 7\}$. The specific features selected by both methods for each dataset are presented in Table 5.

Table 5. The feature sets selected by the examined methods for each data set.

Dataset	Features Selection method	Selected Features
Global	RFE	Pressure, Relative_Wind_Speed, Relative_Humidity, Surface_Temp, Air_Density
	US (KBEST)	Pressure, Relative_Wind_Dir, Relative_Wind_Speed, Relative_Humidity, Solar_Radiation
Daytime	RFE	Air_Temp, Relative_Wind_Speed, Relative_Humidity, Surface_Temp, Air_Density
	US (KBEST)	Pressure, Relative_Wind_Dir, Relative_Wind_Speed, Relative_Humidity, Solar_Radiation
Nighttime	RFE	Air_Temp, Relative_Wind_Speed, Relative_Humidity, Surface_Temp_C, Air_Density.
	US (KBEST)	Pressure, Relative_Wind_Dir, Relative_Wind_Speed, Relative_Humidity, Surface_Temp

4.3. Point prediction results

To validate the effectiveness of the framework, we compared the prediction results of the independent models related to the daytime and nighttime shifts against the global model separately. The Nighttime models were tested on the night time testing records, and the day time models were tested for the daytime records. Moreover, the global models were tested for testing records of the daytime dataset, nighttime dataset, and global records dataset. The simulation results, including the evaluation indices RMSE, MAE, MBE, and MAPE are shown in Table 6. Furthermore, we graphically demonstrated the advantage of the proposed prediction approach by showing the prediction curves of the examined models in Figures 7–11.

First, Figure 7 depicts the performance of the global model on both, the whole set of features and the selected set of features provided by the two examined methods for features selection. Figures 8 and 9 depict the performance of each model with the original and selected set of features for the related time shift.

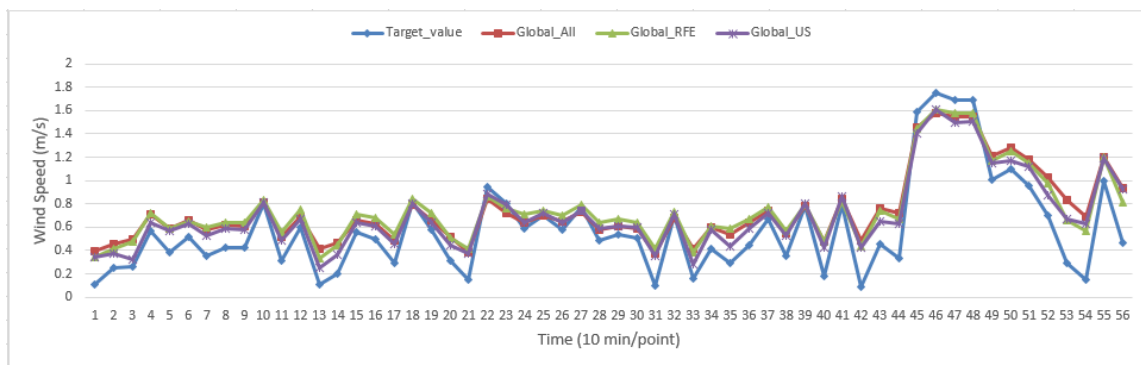


Figure 7. Predictions of global model on global dataset with entire and a selected subset of features.

Table 6. Deterministic 10 minutes ahead statistical prediction scores of WS forecasting models.

Model	Features	Scores			
		RMSE	MAE	MBE	MAPE
SVR-Global	Original (all)	0.1961	0.1556	0.0162	41.9912
	RFE	0.1920	0.1526	0.0214	42.5170
	US-KBest	<u>0.1864</u>	<u>0.1453</u>	<u>0.0015</u>	<u>41.6977</u>
SVR-DT	Original (all)	0.1903	0.1505	0.0243	42.9123
	RFE	0.1925	0.1553	0.0344	41.3541
	US-KBest	<u>0.1821</u>	<u>0.1431</u>	<u>0.0028</u>	<u>41.1200</u>
SVR-NT	Original (all)	0.1892	0.1454	-0.0222	41.5372
	RFE	0.1892	0.1463	-0.0101	40.4290
	US-KBest	<u>0.1801</u>	<u>0.1405</u>	<u>-0.0011</u>	<u>39.6286</u>

The underlined scores in Table 6 designate the best score of a predictor for the three testing datasets examined by the same model. On the other hand, the highlighted scores designate the two best scores of a particular metric across all the models for all the examined testing datasets. The obtained scores show that the nighttime and daytime models outperformed the global models for most of the

adopted evaluation metrics through the subsets of features selected by the US selection method. The RMSE scores designate a good short term performance of the top models. Moreover, the MBE designates the average value of the prediction deviation. The global models provided comparable MBE scores when they were examined on a subset of selected features chosen by the US method.

The daytime model SVR-DT with features selected by US method provides a 7.14% decrease of the RMSE given by the SVR-Global model. On the other hand, the nighttime model SVR-NT with features also selected by the US method provides an 8.2% decrease in the RMSE given by the SVR-Global model as shown in Eq (19). Furthermore, the SVR-DT and SVR-NT models decreased the MAE of the global model by 8.0% and 9.7% when they were applied to the US-Kbest selected dataset.

$$Dec\ in\ RMSE\ \% = 8.2\% = \frac{0.1961 - 0.1801}{0.1961} * 100 \quad (20)$$

On the other hand, the time shift-dependent models showed lower calculated MAPE scores, especially when using selected features. This indicates that these models provide the highest accuracies in terms of the percentage error of their predictions relative to real values. In practical terms, a lower MAPE demonstrates that the predictions of a particular model are closer to the exact values on average, indicating that such a model is reliable.

The scores shown in Table 6 demonstrate that the US-Kbest features selection method over-performed the RFE models when applied with the adopted type of ML models in our framework.

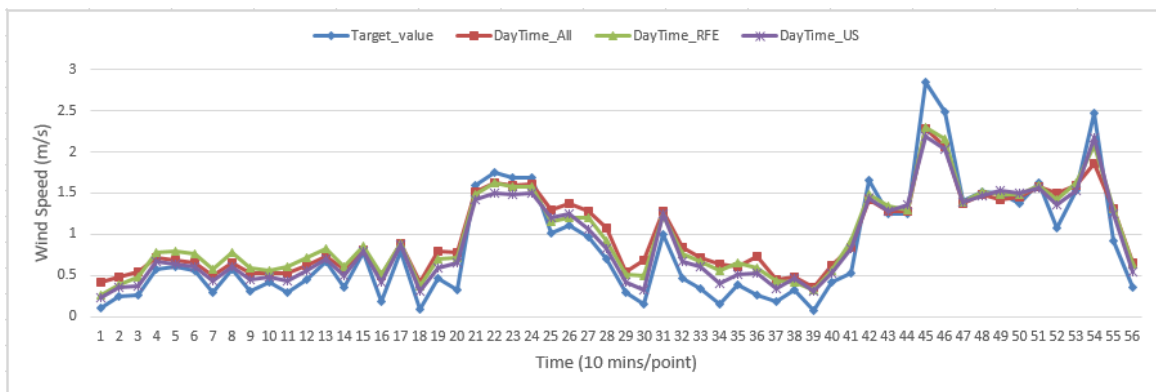


Figure 8. Predictions of day model on daytime dataset with entire and selected features datasets.

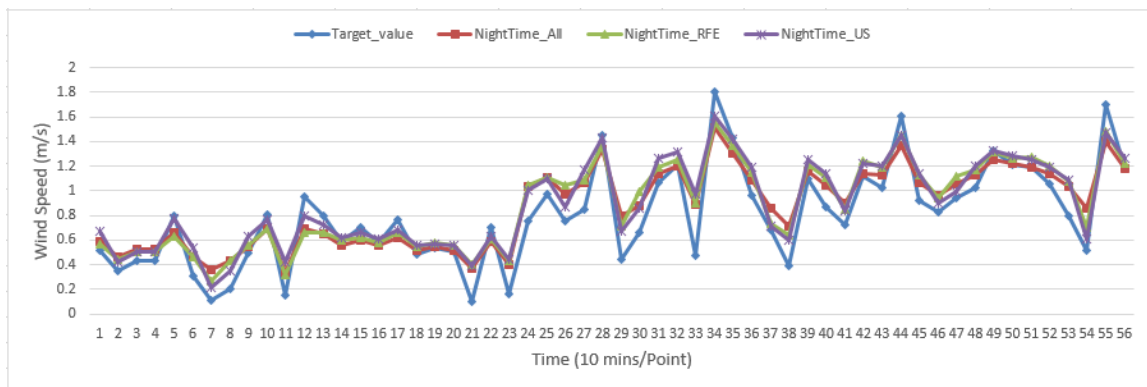


Figure 9. Predictions of night model on the Nighttime dataset with entire and selected feature datasets.

Figures 8 and 9 show that the SVR-DT and SVR-NT models show better overlapping with target values.

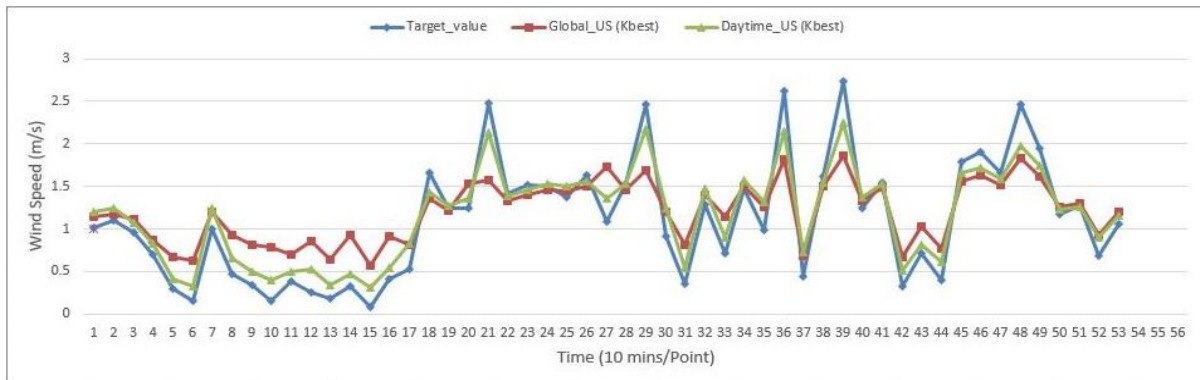


Figure 10. The predictions of global model vs. daytime model with selected features provided by US- Kbest method.

Figure 10 illustrates the curve of the global model versus the daytime model with features selected by the US method for a particular daytime period. Besides, Figure 11 illustrates the curve of the global model versus the nighttime model with features selected by the US method for a specific daytime period.

The comparative charts in Figure 11 and Figure 12 demonstrate that the proposed framework exhibits good point forecast performances against the reference global model that does not differentiate between the daytime and nighttime records.

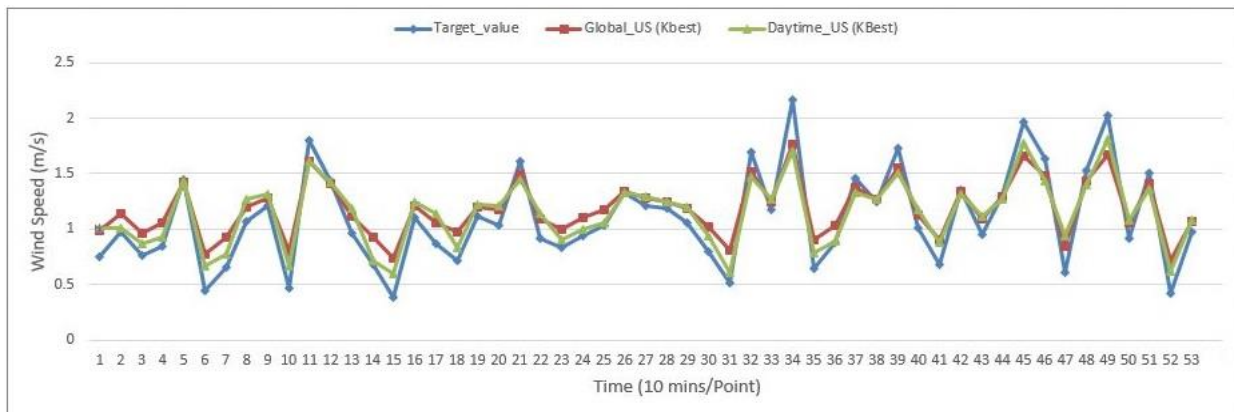


Figure 11. The predictions of the global model vs. the nighttime model with selected features provided by the US- Kbest method.

4.4. Statistical hypotheses testing of the results

Statistical hypothesis testing, specifically paired sample t-tests, was conducted to compare the target values (real values) with the predicted values obtained through different approaches: Full

Features, REF, and US-selected features. The analysis results are briefed in Table 7.

The hypothesis tests above assess the similarity between the average target value (actual real value) and the averages predicted by three different forecasting methods: all features, REF, and US, at the 5% significance level. The null hypothesis states that there is no difference between the averages of the methods, while the alternative hypothesis asserts that there are significant differences.

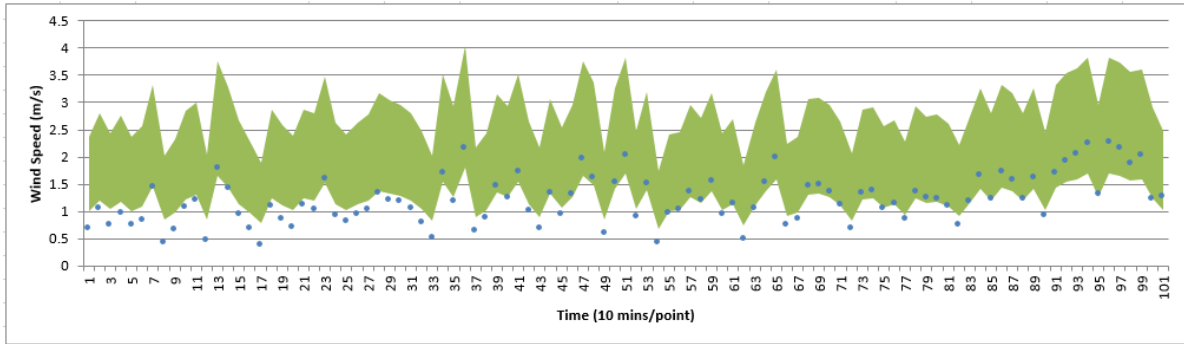
The obtained results indicate that forecasting using the US method is not significantly different from the actual real value. Moreover, the difference between the real mean value and the forecasted mean value using the US method is the smallest, highlighting that the US method closely predicts the actual real value. Similarly, the all-features model also shows no significant difference from the real value, though it exhibits a larger deviation compared to the US method.

In terms of correlations with the actual value, the US method achieves the highest correlation (0.874), outperforming both the all features (0.840) and REF (0.831) methods. This further emphasizes the accuracy of the US method in forecasting the target value.

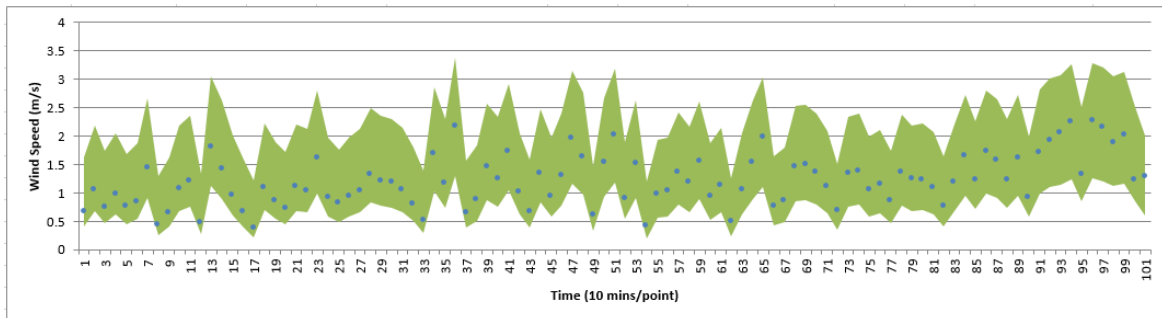
The superior performance of the US method, despite using fewer features compared to the all-features method, highlights the importance of the sensitivity of the features included in the US method. This suggests that the features omitted in the US method are less critical for accurately predicting the real value.

Table 7. Results of statistical hypothesis testing.

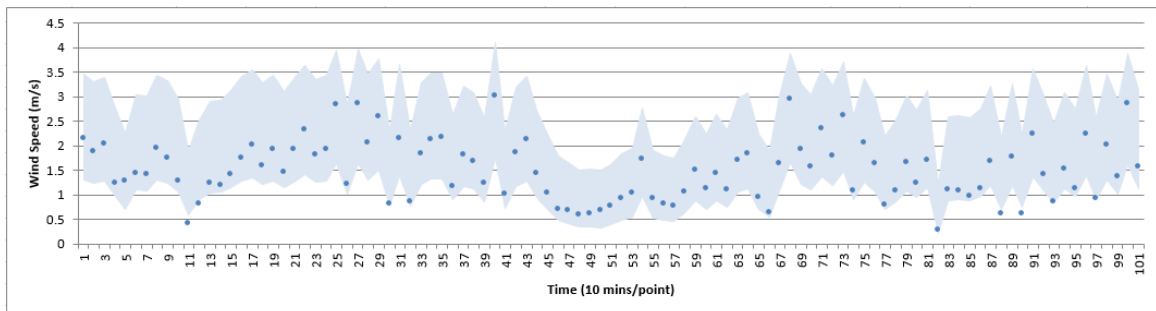
		Mean	Std. Dev.	Std. Error Mean	95% Confidence Interval of the Difference		z-value	df.	P-value
					Lower	Upper			
Pair 1	Target_values vs. pred. (all features)	-0.00696	0.35756	0.00532	-0.01741	0.00347	-1.308	4503	0.191
Pair 2	Target_values vs. pred. (RFE features)	-0.02963	0.34991	0.00521	-0.03985	-0.01941	-5.684	4503	.000
Pair 3	Target values vs. pred (US features)	-0.00077	0.34280	0.00510	-0.01078	0.00924	-0.151	4503	0.880
Pair 4	pred. (all features) vs. pred. (RFE features)	-0.02266	0.08593	0.00128	-0.02517	-0.02015	-17.701	4503	0.000
Pair 5	pred. (all features) vs. pred. (US features)	0.00619	0.10514	0.00156	0.00312	0.00927	3.957	4503	0.000
Pair 6	pred. (RFE features) vs. pred. (US features)	0.02886	0.11356	0.00169	0.02554	0.03218	17.058	4503	0.000



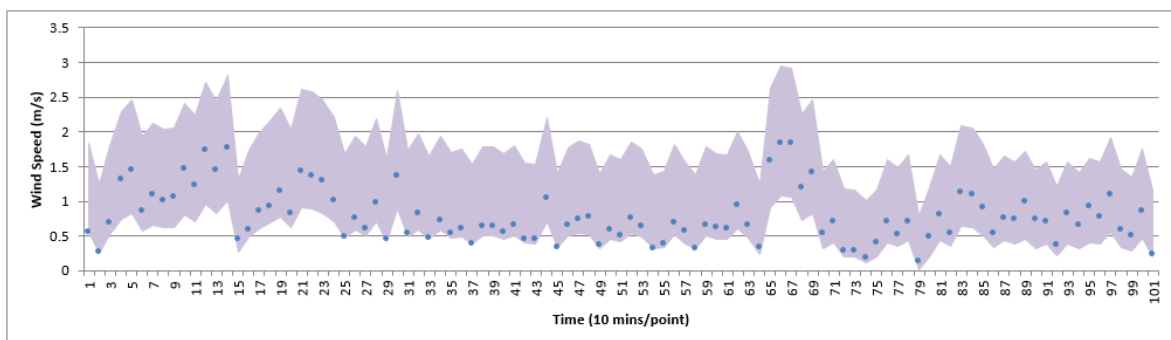
(a) Global model with all features (85% confidence level)



(b) Global model with selected features (85% confidence level)



(c) SVR-DT model with selected features (85% confidence level)



(d) SVR-NT model with selected features (85% confidence level)

Figure 12. Prediction intervals of wind speed forecasting provided by global, daytime, and nighttime models with 85% confidence level.

4.5. Interval prediction results

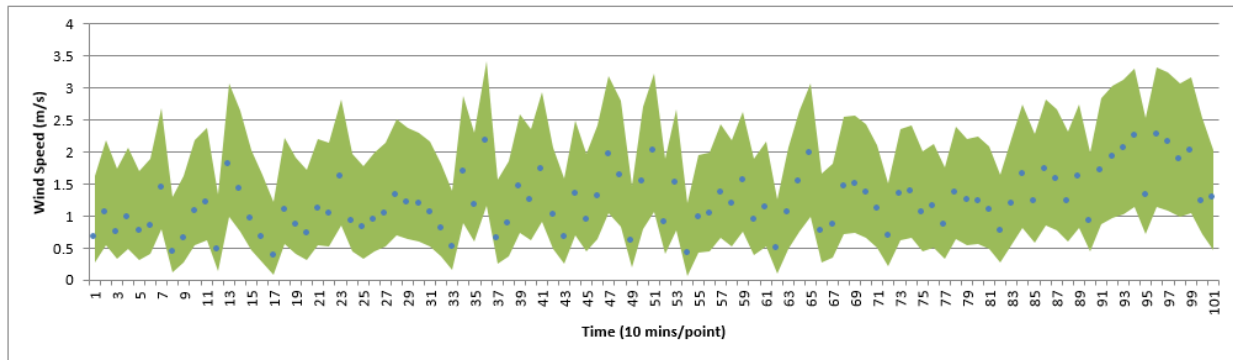
In this section, we assess the efficiency and performance of the proposed probabilistic method in forecasting the PIs of WS. The simulations estimated the probability density distribution and the related CDFs of prediction errors for each predictor described in Section 3.3 with different confidence levels. Table 8 depicts the evaluation indices of the PI forecasting provided by the examined models with confidence levels of 85%, 90%, and 95%, respectively. For clarity, we consider only the models that are trained on all features and those trained on features selected by the US method. The best PICP scores (close to the confidence level) for each confidence level are bolded, and the best related CWC score is underlined.

Table 8. The scores obtained for the evaluation indices of the PIs forecasting with several PINC.

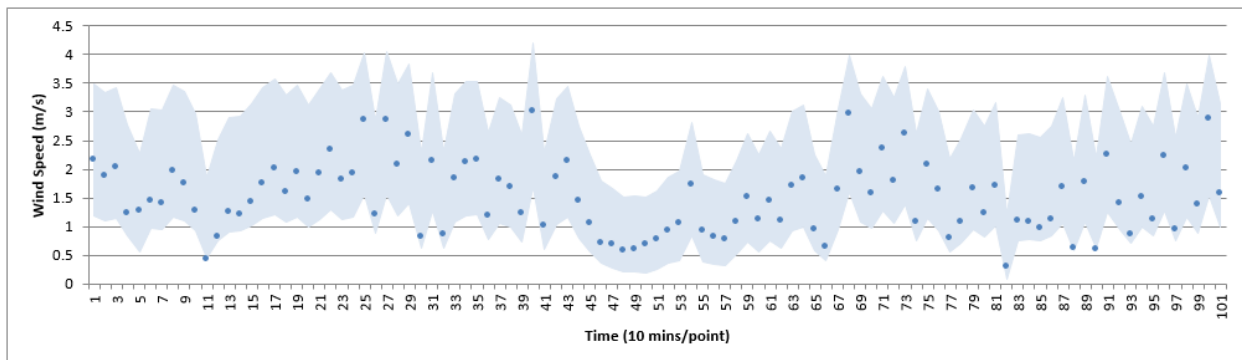
Model	Features	PINC = 85% ($\alpha = 0.15$)			PINC = 90% ($\alpha = 0.10$)			PINC = 95% ($\alpha = 0.05$)		
		PICP	PINAW	CWC	PICP	PINAW	CWC	PICP	PINAW	CWC
SVR-Global	ALL	0.4304	0.1506	9.1548	0.9489	0.4015	0.4015	0.9959	0.5492	0.5492
	US-Kbest	0.8908	0.2964	0.2964	0.9509	0.4002	0.4002	0.9859	0.5546	0.5546
SVR-DT	ALL	0.7387	0.2773	1.1210	0.8978	0.3681	0.7441	0.9859	0.5546	0.5546
	US-Kbest	0.8568	0.2707	<u>0.2707</u>	0.9339	0.3634	0.3634	0.9749	0.4996	<u>0.4996</u>
SVR-NT	ALL	0.8518	0.3203	0.3203	0.9199	0.3927	0.3927	0.9829	0.6003	0.6003
	US-Kbest	0.8428	0.2843	0.4298	0.9119	0.3959	<u>0.3959</u>	0.9801	0.5317	0.5317

Table 8 shows that the independent time shift models outperform the reference global model in almost all the evaluation indices within most of the confidence levels. The best obtained scores are provided by the time shift-dependent models, particularly those utilizing selected features for each particular time shift. For PINC = 85%, the SVR-DT model (with selected features) and the SVR-NT model showed the best-scored PICP with a competitive CWC score provided by the SVR-DT model. Also, for PICP = 90%, the two independent models show the best scores against the global model. As per PICP = 95%, the two independent models coupled with the features selection method showed the closest scores to the target confidence level with the lowest CWC scores. Due to the inherent trade-off between the PICP and PINAW evaluation metrics, there are some cases where only one evaluation index of a predictor is better than that of another, e.g., the case of the best models for $PICP = 95\%$. In other cases, we can see that the PICP scores of the independent models are slightly better than those of the global reference model, but the scores of the comprehensive evaluation index CWC of the latter ones are, in general, smaller than those the former ones. For instance, at the confidence level of 85%, the CWC index of the SVR-DT model (0.2707) is smaller than the CWC score of the SVR-Global with all features (9.1584) and even smaller than the SVR-Global (0.2964) that uses the same feature selection method. Also, at the confidence level of 95%, the CWC index of the SVR-DT model (0.4996) is smaller than the CWC score of the SVR-Global with all features (0.5492) and even smaller than the SVR-Global (0.5546) that uses the same feature selection method. On the other hand, at the confidence level of 90%, the CWC index of the SVR-NT model (0.3959) is better than both SVR-Global models.

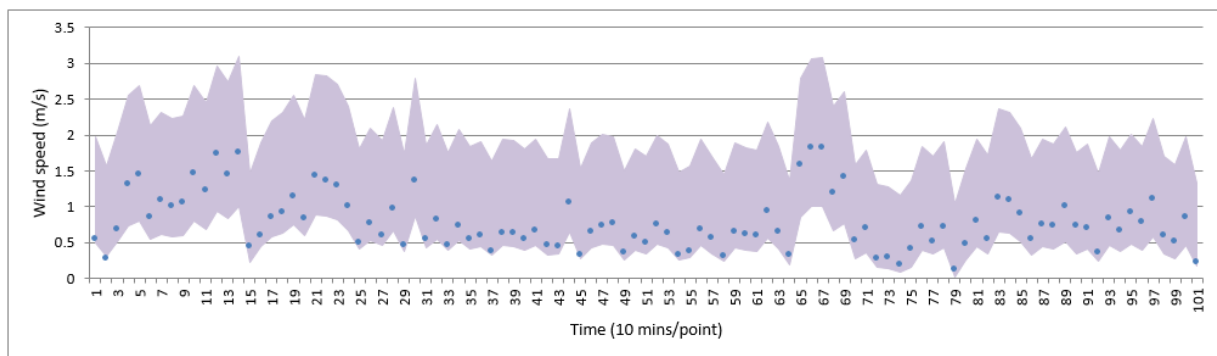
Considering that the CWC accounts for both PICP and PINAW, this study showed that our proposal of handling the daytime and nighttime weather data records for PI estimation of WS demonstrates outstanding forecasting performance.



(a) Global model with selected features (90% confidence level)



(b) SVR-DT model with selected features (90% confidence level)

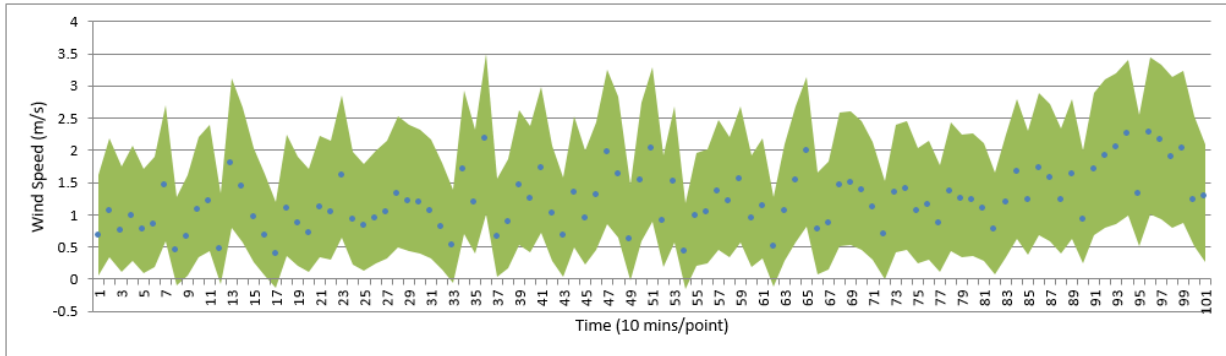


(c) SVR-NT model with selected features (90% confidence level)

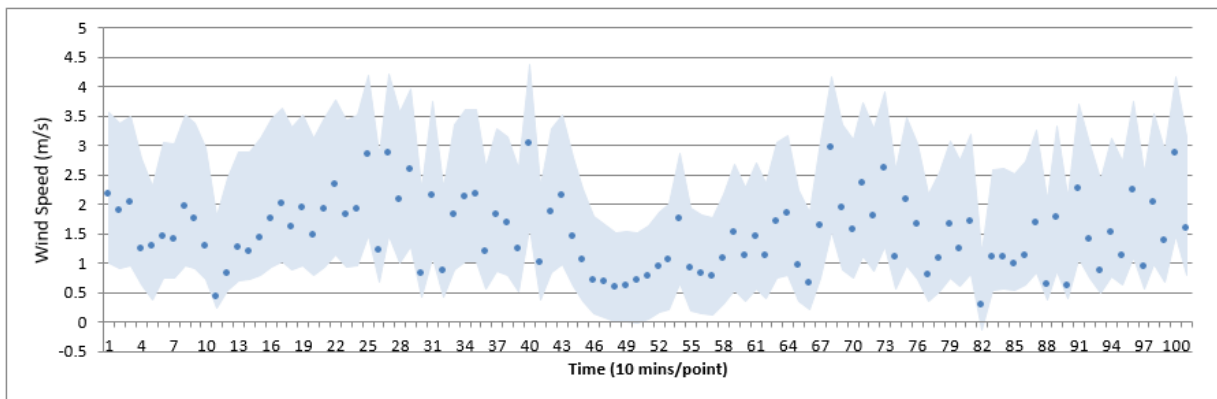
Figure 13. Prediction intervals of wind speed forecasting provided by global, daytime, and nighttime models with 90% confidence level.

Figures 12–14 show the PI estimation results of the best models with a randomly selected period within different confidence levels, i.e., 85%, 90%, and 95%, respectively.

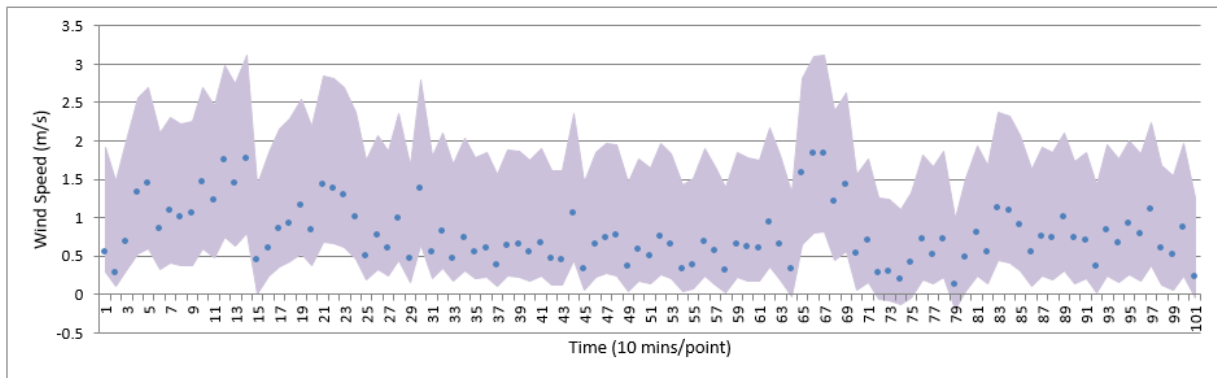
The PIs constructed by the independent time shift models show a good coverage rate with competitive interval widths. The overall performance indicates that the introduced prediction framework can construct high-quality PIs for testing wind speed datasets at various confidence levels.



(a) Global model with selected features (95% confidence level)



(b) SVR-DT model with selected features (90% confidence level)



(c) SVR-DT model with selected features (95% confidence level)

Figure 14. Prediction intervals of wind speed forecasting provided by global, daytime, and nighttime models with a 95% confidence level.

5. Conclusions

Wind energy is increasingly gaining attention from scholars and industries. However, accurate short-term WS prediction is a challenging task when controlling the consumption/production of wind energy within a given smart grid. Related works emphasize the necessity of carrying out uncertainty wind speed modeling, which provides not only the prediction error but also the probability of correct

predictions. Practically, reliable PI estimators of WS are crucial to the evaluation and risk analysis of wind power for decision-makers.

We present a comprehensive prediction framework for short-term WS PI forecasting using ML. The framework consists of four modules for data processing, feature selection, point prediction, and interval forecasting of WS. The data processing module splits data records into daytime and nighttime subsets and then employs the QM method to identify and eliminate outliers. Independent daytime and nighttime SVR-based predictors with RBF kernels were trained and validated for WS point predictions.

The PI forecasting module employed the nonparametric KDE method to analyze and estimate the PDFs of the point prediction errors of each time-shift-related predictor. The estimated PDFs and the CDFs were used to compute the fluctuation range of the prediction error given a PI confidence level, obtaining different interval prediction results. The proposed WS prediction framework is validated for short-term WS forecasting using a weather dataset that consists of nine weather and wind speed parameters. Simulation results indicate that considering daytime and nighttime data records and applying suitable feature selection along with independent ML models provides better forecast accuracy than using global prediction models. For PI estimation, the evaluation indices of the proposed interval forecasting models are consistently smaller than those of the global ones. The main contribution of this work is to suggest and validate a framework for wind speed PI forecasting that may include other ML regression models, not necessarily the SVR ones.

Some limitations of the designed PI framework need to be addressed in our future research works. For instance, in the current framework, we assume that all deterministic prediction models are of the same type and have the same structure, namely SVR. In addition, the bandwidth parameter h , which determines the width of the distribution interval of the prediction error and thereby controls the smoothness of the estimation, is selected using the trial and error method. Other future perspectives include validating the proposed approach on different time horizons and resolutions, and enhancing prediction performance while improving operating time.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

Conceptualization, R. A. H., G. O., and A. A.; methodology, R. A. H., M. F., G. O., and A.A.; software, R. A. H., M. F.; validation, R. A. H., G. O., and M. F.; formal analysis, R.A. H. and G. O.; data preparation, R. A. H., and G. O.; writing the first draft, R. A. H., M. F., and A. A.; review and proofreading, G. O., A. A.; visualization, R. A. H., and M. F.; simulations management, R. A. H., and G. O.

Conflict of interest

The authors declare there is no conflicts of interest.

References

1. S. Roga, S. Bardhan, Y. Kumar, S. K. Dubey, Recent technology and challenges of wind energy generation: A review, *Sustainable Energy Technol. Assess.*, **52** (2022), 102239. <https://doi.org/10.1016/j.seta.2022.102239>
2. T. M. Dinku, M. S. Manshahia, K. S. Chahal, Soft computing techniques for maximum power point tracking in wind energy harvesting system: A survey, in *Artificial Intelligence for Renewable Energy and Climate Change*, (2022), 137–170. <https://doi.org/10.1002/9781119771524.ch6>
3. Y. Zhuo, L. Li, J. Tang, W. Meng, Z. Huang, K. Huang, et al., Optimal real-time power dispatch of power grid with wind energy forecasting under extreme weather, *Math. Biosci. Eng.* **20** (2023), 14353–14376. <https://doi.org/10.3934/mbe.2023642>
4. S. M. R. H Shawon, M. A. Saaklayen, X. Liang, Wind speed forecasting by conventional statistical methods and machine learning techniques, in *2021 IEEE Electrical Power and Energy Conference (EPEC)*, (2021), 304–309. <https://doi.org/10.1109/EPEC52095.2021.9621686>
5. L. Peng, S. X. Lv, L. Wang, Explainable machine learning techniques based on attention gate recurrent unit and local interpretable model-agnostic explanations for multivariate wind speed forecasting, *J. Forecast.*, **43** (2024), 064–2087. <https://doi.org/10.1002/for.3097>
6. Y. Yang, H. Lou, J. Wu, S. Zhang, S. Gao, A survey on wind power forecasting with machine learning approaches, *Neural Comput. Appl.*, (2024), 1–21. <https://doi.org/10.1007/s00521-024-09923-4>
7. Y. Yang, Y. Gao, Z. Wang, X. Li, H. Zhou, J. Wu, Multiscale-integrated deep learning approaches for short-term load forecasting, *Int. J. Mach. Learn. Cybern.*, **15** (2024), 6061–6076.
8. Y. L. Chen, X. Hu, L. X. Zhang, A review of ultra-short-term forecasting of wind power based on data decomposition-forecasting technology combination model, *Energy Reports*, **8** (2022), 14200–14219. <https://doi.org/10.1016/j.egy.2022.10.342>
9. R. A. Hajj, M. M. Fouad, A. Assi, E. Mabrouk, Ultra-short-term forecasting of wind speed using lightweight features and machine learning models, in *2023 12th International Conference on Renewable Energy Research and Applications (ICRERA)*, (2023), 93–97. <https://doi.org/10.1109/ICRERA59003.2023.10269374>
10. Z. Wang, Y. Ying, L. Kou, W. Ke, J. Wan, Z. Yu, et al., Ultra-short-term offshore wind power prediction based on PCA-SSA-VMD and BiLSTM, *Sensors*, **24** (2024), 444. <https://doi.org/10.3390/s24020444>
11. J. Naik, R. Bisoi, P. K. Dash, Prediction interval forecasting of wind speed and wind power using modes decomposition based low rank multi-kernel ridge regression, *Renewable Energy*, **129** (2018), 357–383. <https://doi.org/10.1016/j.renene.2018.05.031>
12. R. Li, Y. Jin, A wind speed interval prediction system based on multi-objective optimization for machine learning method, *Appl. Energy*, **228** (2018), 2207–2220. <https://doi.org/10.1016/j.apenergy.2018.07.032>
13. W. Ding, F. Meng, Point and interval forecasting for wind speed based on linear component extraction, *Appl. Soft Comput.*, **93** (2020), 106350. <https://doi.org/10.1016/j.asoc.2020.106350>
14. H. Huang, Y. Hong, H. Wang, Probabilistic prediction intervals of wind speed based on explainable neural network, *Front. Energy Res.*, **10** (2022), 934935. <https://doi.org/10.3389/fenrg.2022.934935>

15. J. Zhang, C. Draxl, T. Hopson, L. DelleMonache, E. Vanvyve, B. M. Hodge, Comparison of numerical weather prediction based deterministic and probabilistic wind resource assessment methods, *Appl. Energy*, **156** (2015), 528–541. <https://doi.org/10.1016/j.apenergy.2015.07.059>
16. G. Hou, J. Wang, Y. Fan, J. Zhang, C. Huang, A novel wind power deterministic and interval prediction framework based on the critic weight method, improved northern goshawk optimization, and kernel density estimation, *Renewable Energy*, **226** (2024), 120360. <https://doi.org/10.1016/j.renene.2024.120360>
17. W. Yang, M. Hao, Y. Hao, Innovative ensemble system based on mixed frequency modeling for wind speed point and interval forecasting, *Inf. Sci.*, **622** (2023), 560–586. <https://doi.org/10.1016/j.ins.2022.11.145>
18. Z. Tian, J. Wang, A novel wind speed interval prediction system based on neural network and multi-objective grasshopper optimization, *Int. Trans. Electr. Energy Syst.*, **1** (2022), 5823656. <https://doi.org/10.1155/2022/5823656>
19. X. Wang, J. Wang, X. Niu, C. Wu, Novel wind-speed prediction system based on dimensionality reduction and nonlinear weighting strategy for point-interval prediction, *Expert Syst. Appl.*, **241** (2024), 122477. <https://doi.org/10.1016/j.eswa.2023.122477>
20. Y. Zhang, Y. Zhao, X. Shen, J. Zhang, A comprehensive wind speed prediction system based on Monte Carlo and artificial intelligence algorithms, *Appl. Energy*, **305** (2022), 117815. <https://doi.org/10.1016/j.apenergy.2021.117815>
21. J. Wang, S. Wang, B. Zeng, H. Lu, A novel ensemble probabilistic forecasting system for uncertainty in wind speed, *Appl. Energy*, **313** (2022), 118796. <https://doi.org/10.1016/j.apenergy.2022.118796>
22. Q. Zhu, Y. Xu, Q. Lin, Z. Ming, K. C. Tan, Clustering-based short-term wind speed interval prediction with multi-objective ensemble learning, *IEEE Trans. Emerging Topics Comput. Intell.*, (2024). <https://doi.org/10.1109/TETCI.2024.3400852>
23. P. Sun, Z. Liu, J. Wang, W. Zhao, Interval forecasting for wind speed using a combination model based on multi-objective artificial hummingbird algorithm, *Appl. Soft Comput.*, **150** (2024), 111090. <https://doi.org/10.1016/j.asoc.2023.111090>
24. G. Tang, Y. Wu, C. Li, P. K. Wong, Z. Xiao, X. An, A novel wind speed interval prediction based on error prediction method, *IEEE Trans. Ind. Inf.*, **16** (2020), 6806–6815. <https://doi.org/10.1109/TII.2020.2973413>
25. A. Saeed, C. Li, M. Danish, S. Rubaiee, G. Tang, Z. Gan, et al., Hybrid bidirectional LSTM model for short-term wind speed interval prediction, *IEEE Access*, **8** (2020), 182283–182294. <https://doi.org/10.1109/ACCESS.2020.3027977>
26. Y. Zhang, G. Pan, Y. Zhao, Q. Li, F. Wang, Short-term wind speed interval prediction based on artificial intelligence methods and error probability distribution, *Energy Convers. Manage.*, **224** (2020), 113346. <https://doi.org/10.1016/j.enconman.2020.113346>
27. J. Wang, S. Wang, Z. Li, Wind speed deterministic forecasting and probabilistic interval forecasting approach based on deep learning, modified tunicate swarm algorithm, and quantile regression, *Renewable Energy*, **179** (2021), 1246–1261. <https://doi.org/10.1016/j.renene.2021.07.113>
28. Z. Gan, C. Li, J. Zhou, G. Tang, Temporal convolutional networks interval prediction model for wind speed forecasting, *Electr. Power Syst. Res.*, **191** (2021), 106865. <https://doi.org/10.1016/j.epsr.2020.106865>

29. W. Ding, F. Meng, Point and interval forecasting for wind speed based on linear component extraction, *Appl. Soft Comput.*, **93** (2020), 106350. <https://doi.org/10.1016/j.asoc.2020.106350>
30. C. Li, G. Tang, X. Xue, X. Chen, R. Wang, C. Zhang, The short-term interval prediction of wind power using the deep learning model with gradient descend optimization, *Renewable Energy*, **155** (2020), 197–211. <https://doi.org/10.1016/j.renene.2020.03.098>
31. M. R. Islam, A. A. Lima, S. C. Das, M. F. Mridha, A. R. Prodeep, Y. Watanobe, A comprehensive survey on the process, methods, evaluation, and challenges of feature selection, *IEEE Access*, **10** (2022), 99595-99632. <https://doi.org/10.1109/ACCESS.2022.3205618>
32. P. Agrawal, H. F. Abutarboush, T. Ganesh, A. W. Mohamed, Metaheuristic algorithms on feature selection: A survey of one decade of research (2009–2019), *IEEE Access*, **9** (2021), 26766–26791. <https://doi.org/10.1109/ACCESS.2021.3056407>
33. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, et al., Scikit-learn: Machine learning in Python, *J. Mach. Learn. Res.*, **12** (2011), 2825–2830.
34. M. Awad, R. Khanna, M. Awad, R. Khanna, Support vector regression, in *Efficient Learning Machines*, Springer, (2015), 67–80. https://doi.org/10.1007/978-1-4302-5990-9_4
35. B. Kumar, O. P. Vyas, R. A. Vyas, Comprehensive review on the variants of support vector machines, *Modern Phys. Lett. B*, **33** (2019), 1950303. <https://doi.org/10.1142/S0217984919503032>
36. J. Wu, Y. G. Wang, H. Zhang, Augmented support vector regression with an autoregressive process via an iterative procedure, *Appl. Soft Comput.*, **158** (2024), 111549. <https://doi.org/10.1016/j.asoc.2024.111549>
37. Y. C. Chen, A tutorial on kernel density estimation and recent advances, *Biostat. Epidemiol.*, **1** (2017), 161–187. <https://doi.org/10.1080/24709360.2017.1396742>
38. S. Węglarczyk, Kernel density estimation and its application, in *ITM Web of Conferences*, **23** (2018), 00037. <https://doi.org/10.1051/itmconf/20182300037>
39. H. Wang, Z. Lei, X. Zhang, B. Zhou, J. Peng, A review of deep learning for renewable energy forecasting, *Energy Convers. Manage.*, **198** (2019), 111799. <https://doi.org/10.1016/j.enconman.2019.111799>



AIMS Press

©2024 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0>)