A Novel Hybrid Neural Network-Based Day-Ahead Wind Speed Forecasting Technique

Mehdi Abbasipour, Student Member, IEEE, Mosayeb Afshari Igder, Student Member, IEEE, and Xiaodong Liang, Senior Member, IEEE
Department of Electrical and Computer Engineering, University of Saskatchewan, Saskatoon, SK S7N 5A9 Canada
Corresponding author: Xiaodong Liang (e-mail: xil659@mail.usask.ca).
This work was supported in part by the Natural Science and Engineering Research Council of Canada (NSERC) Discovery Grant RGPIN-2016-04170.

ABSTRACT As a dominant form of renewable energy sources with significant technical progress over the past decades, wind power is increasingly integrated into power grids. Wind is chaotic, random and irregular. For proper planning and operation of power systems with high wind power penetration, accurate wind speed forecasting is essential. In this paper, a novel hybrid Neural Network (NN)-based day-ahead (24 hour horizon) wind speed forecasting is proposed, where five hybrid neural network algorithms are evaluated. The five algorithms include Wavelet Neural Network (WNN) trained by Improved Clonal Selection Algorithm (ICSA), WNN trained by Particle Swarm Optimization (PSO), Extreme Learning Machine (ELM)-based neural network, Radial Basis Function (RBF) neural network, and Multi-Layer Perceptron (MLP) Neural Network. Single- and multi-features and their effect on the accuracy of wind speed prediction are also analyzed. The wind speed dataset used in this paper is Saskatchewan's recorded historical wind speed data. Despite the excellent wind power potential, only 6.5% of the total electricity demand is currently supplied by wind power in Saskatchewan, Canada. This study paves the way for economical operation, planning, and optimization of Saskatchewan's future wind power generation.

INDEX TERMS Hybrid Neural Network, Machine Learning, Day-ahead Wind Speed Forecasting, Wind Power.

I. INTRODUCTION
Wind power plays an increasing role in the modern mixed energy landscape by producing sustainable clean energy and reducing fossil fuel-based conventional power generation. Wind turbines have been installed in windy regions, such as Saskatchewan in Canada, which benefit from ample wind resources due to their geographical locations and terrain features. In Saskatchewan, the currently installed wind power capacity only approximately provides 6.5% of the electricity demand, and the province has the plan to boost wind power up to 40% of the demand by 2030 [1]. However, one challenge associated with wind power is intermittent wind speed. The accurate wind speed prediction methods are urgently needed to improve economical, secure, and reliable operation of wind power systems [2].

The wind speed prediction can be performed in five time scales [3]–[5]: very short-term prediction (a few seconds to 30 minutes ahead); short-term prediction (30 minutes to 6 hours ahead); medium-term prediction (6 hours to 24 hours ahead); long-term prediction (24 hours to one week ahead); and very long-term prediction (one week and longer) [6]. The applications of each time-scale are provided in [3]–[5]. To realize wind speed forecasting, the following four types of methods can be used: physical or weather-based methods; statistical or time-series-based methods; Artificial Intelligence (AI)-based methods; and hybrid methods [4], [5], [7].

Physical or weather-based wind speed prediction methods use the meteorological data of wind power plants. The utilized data include essential factors, such as the land topology, atmospheric temperature, pressure, humidity, obstacles, and surface coarseness. The predicted wind speeds are used with power curves of wind turbines to derive wind power generation. However, these methods lead to high computational burden, and thus, require a long processing time [4].

Statistical or time-series wind speed prediction methods are data-driven approaches utilizing historical wind speed data, and the one-step statistical analysis [7]. Auto Regression Moving Average (ARMA) and Auto Regression Integrated Moving Average (ARIMA) are two common statistical
methods being used [8], [9]. The ARIMA-based models can predict wind speed efficiently for short-term applications [9], but their performance is not acceptable for long-term applications, and thus, more improvements are needed. Fractional ARIMA (FARIMA) and Markov chain (with its robustness in sequential data modeling) methods are classified as statistical methods [10], [11]. In [10], the FARIMA method can provide effective results compared to conventional methods for long-term wind speed forecasting, especially for sites with severe intermittent wind speed. Statistical methods are suitable for short-term applications, and their implementation is very easy and straightforward [11], but the issue is they are linear-based predictors, while wind speed is a nonlinear phenomenon. In [12], the Markov chain is employed for short- and medium-term wind speed forecasting.

AI-based wind speed prediction methods, such as Artificial Neural Network (ANN) and Support Vector Machines (SVM), are data-driven and multi-step approaches. Unlike statistical methods (which are linear approaches), AI-based methods are suitable for nonlinear forecasting. In the literature, ANN is primarily used in the short-term wind speed forecasting. Characteristics of the wind speed forecasting problem and considerations of the analysis are two important factors in determining a proper ANN model and the number of neurons [4]. The principal of ANN techniques lies in mapping actual nonlinear input data into forecasted values. Therefore, a suitable training approach is chosen to train the ANN-based model to reach proper weight values based on the recorded data, i.e., an ANN model learns from the gained experience through the training procedure, then uses nonlinear input data to forecast wind speed [13]. A proper number of neurons should be employed depending on the nature of wind speed forecasting problem. By improving the general ANN techniques, new ANN methods are proposed for wind speed forecasting, including Radial Basis Function (RBF) neural networks, Recurrent Neural Networks (RNNs), Feed-Forward Back Propagation (FFBP) neural networks, and Multi-Layer Perceptron (MLP) neural networks [4], [5], [14], [15]. Ref. [16] reveals that RBF and FFBP neural networks are effective in improving ANN's performance by considering the nonlinearity of wind speed. In [16], SVM and MLP neural network are compared for wind speed forecasting, and SVM outperforms MLP neural network in most cases. In [17], five ANN methods are used for short-term wind speed forecasting, among them, the ANN consisting of 19 hidden layers, 4 input layers, and one output layer has the best performance, while a single ANN with a traditional training technique has the limited capability to follow the nonlinear behavior of wind speed.

Hybrid wind speed prediction methods are data-driven approaches and introduced to tackle the shortcomings of physical, statistical, and AI-based methods. Hybrid methods usually include two or more prediction methods, or they can be filter-augmented methods. Filter-augmented methods use a special filter added to a basic method to improve the forecasting performance. Kalman Filter (KF) and Wavelet Transform are two popular filters added to some statistical (ARIMA) and AI-based methods (ANN). In [18], KF and ANN methods are added to the ARIMA method to create the hybrid KF-ANN-ARIMA method. The obtained results reveal that the KF-ANN-ARIMA method is more effective, and successfully improves the efficiency and accuracy of the original ARIMA method in wind speed forecasting. Wavelet Transform is added to ANNs to create the Wavelet Neural Network (WNN), and choosing a proper Wavelet Transform affects the accuracy of wind speed prediction using WNN [4].

Statistical methods have been added to AI-based methods to improve wind speed forecasting performance. In [18], [19], three methods, ARIMA, ANN, and ANN-ARIMA, are compared for wind speed prediction in terms of accuracy, and the hybrid ANN-ARIMA method provides a better accuracy in different time scales than ARIMA and ANN, and thus, is a suitable method for datasets with nonlinear and linear patterns. The fuzzy logic concept was added to the self-learning capability of ANN to improve the performance of wind speed forecasting, which leads to the creation of Fuzzy Neural Networks (FNNS), and Adaptive Neuro-Fuzzy Inference System (ANFIS) [4]. Particle Swarm Optimization (PSO) and Whale Optimization Algorithm (WOA) can also be added to ANN. PSO is combined with ANFIS to form the PSO-ANFIS method to improve the performance of the original ANFIS method. In [20], WOA is added to the Support Vector Regression (SVR) method to improve the accuracy and simplify the implication of SVR, as SVR is basically constrained by the penalty factor (C) and kernel parameter (K). It is found that the WOA-SVR method has better performance than the PSO-SVR method. As well-known training algorithms, Imperialistic Competitive Algorithm (ICA), Improved Clonal Selection Algorithm (ICSA), and Extreme Learning Machine (ELM) are also added to hybrid methods [21]–[23].

Among wind speed prediction methods reported in the literature, hybrid methods are more advanced approaches, but their development is only at the beginning stage. Also, in most wind speed prediction work, single and multiple meteorological factors have not been considered and compared. Another issue is the lack of a comprehensive procedure for wind speed forecasting among published papers. Therefore, this paper aims to address the above mentioned technical deficiencies in wind speed forecasting area.

The major contributions of this paper include:

- A novel hybrid neural network-based day-ahead (24 hour horizon) wind speed forecasting technique is proposed in this paper. Five hybrid neural network methods (WNN trained by ICSA; WNN trained by PSO; ELM-based neural network; RBF neural network; and MLP neural network) are evaluated using measured historical wind speed data in Saskatchewan, Canada.
- A comprehensive procedure to implement the proposed technique is developed for wind speed forecasting.
The designed algorithm for the ELM model is completely different. Therefore, this stage consists of several steps considering the employed algorithm/model. The details of each step/algorithm are evaluated below.

As the employed algorithms for WNN, RBF, and MLP models are similar, the related steps for their algorithms are provided as follows:

- **Step 1**: The parameters of the WNN, RBF, and MLP models are initialized.
- **Step 2**: The training method for each model is chosen. The two methods, ICSA and PSO, are considered to train the WNN model. The ICSA method is also considered to train the RBF and MLP models, i.e., ICSA is the default training method for the RBF and MLP models.
- **Step 3**: After training the forecasting system, parameters of the models are updated.
- **Step 4**: In this step, the termination criterion is checked. If the required condition is not met, the algorithms follow the procedure back to Step 2, the system will be re-trained, and parameters will be updated again until meeting the required condition for termination. If the required condition is met, the algorithms move on to Step 5.
- **Step 5**: After reaching the termination criterion, the obtained parameters are considered as the final results for the model parameters, which will be used to form the prediction models.

In the ELM-based model, which uses a single-layer FFNN, the ELM training technique is used to train the neural network and determine its parameters. The related steps of this training method are provided as follows:

- **Step 1**: The ELM method is used to train the neural network.
- **Step 2**: The obtained parameters in Step 1 are considered as the final parameters of the ELM model to form the prediction model.

4) **Stage 4**: This stage is to test the prediction systems and choose the best approach for wind speed forecasting. It consists of four steps as follows.

- **Step 1**: The testing set is applied to the forecasting system.
- **Step 2**: Wind speed is forecasted based on input data. To do so, each sample of wind speed in the testing set is considered as a target, and its previous most relevant data is used to predict wind speed. The autocorrelation function is used here to select the most relevant samples for the considered target.
- **Step 3**: The actual wind speed (the target sample) and the forecasted value using the proposed approach are compared, and the forecasting errors and computational time of the whole procedure are calculated.
- **Step 4**: According to the calculated forecasting errors and computational time, the best model/approach is chosen based on our desired application(s).
Stage 1: Data Processing

- Inputting original wind speed data
- Determining the Training and Testing Sets
- Using Auto Correlation Function for selecting most relevant data in each feature

Stage 2: Employed Models

- WNN
- FFNN
- RBF
- MLP

Stage 3: Training Approaches

- Initializing the parameters
- Choosing Training Approach
- Updating the parameters
- Deriving the final parameters
- Termination?
  - No
  - Yes
- Training the system by ELM
- Deriving the final parameters
- Applying the testing set to the system
- Forecasting wind speed
- Calculating forecasting errors and computational time
- Choosing the best approach by comparing the obtained forecasting errors

Stage 4: Testing

- Initializing the parameters
- Updating the parameters
- Termination?
  - No
  - Yes
- Training the system by ICSA
- Deriving the final parameters
- Applying the testing set to the system
- Forecasting wind speed
- Calculating forecasting errors and computational time
- Choosing the best approach by comparing the obtained forecasting errors

FIGURE 1. The schematic diagram of the proposed approach for day-ahead wind speed forecasting.
III. FUNDAMENTAL THEORIES OF THE EMPLOYED HYBRID NEURAL NETWORK METHODS AND TRAINING APPROACHES

In this paper, five hybrid NN methods based on four basic models, WNN, ELM, RBF, and MLP, are evaluated for day-ahead wind speed forecasting using the proposed technique introduced in Section II. Two training approaches, ICSA and PSO, are chosen in this study for these hybrid models. In this section, fundamental theories of the basic hybrid forecasting methods along with ICSA are briefly explained. The details of the WNN method trained by PSO can be found in [24].

A. FUNDAMENTAL THEORIES OF HYBRID NEURAL NETWORK METHODS

1) Wavelet Neural Network: In the literature, Wavelet Transform is usually used as a preprocessor to decompose time series wind speed to a set of subseries, and to forecast future values of these subseries using statistical or machine learning-based methods; the inverse wavelet transform is then used to construct the original wind speed time series [24]–[26].

In this paper, the wavelet is used to form WNN by considering the wavelet function as an activation function in hidden neurons of the neural network. Mexican hat and Morlet wavelets are two options that can be used as an activation function to predict wind speed [22], [27]. We choose the Morlet wavelet function as shown in Fig. 2. The structure of the WNN is illustrated in Fig. 3. This type of structure is known as a three-layer feed-forward structure, where \( X = [x_1, x_2, \ldots, x_m] \) is the input data, and \( y \) is the model output, known as a target variable.

The activation function in the hidden layer, which is the multi-dimensional Morlet wavelet function, is as follows:

\[
F_i(x_1, x_2, \ldots, x_m) = \prod_{j=1}^{m} \phi \left( \frac{x_j - b_j}{a_j} \right)
\]

(1)

\[
\phi(x) = e^{-0.5x^2} \cos(5x)
\]

(2)

where \( a_i \) and \( b_i \) are scale and shift parameters, respectively. \( \phi(x) \) is the Morlet wavelet function. \( n \) is the number of hidden neurons of the wavelet neural network. Eventually, the output of the WNN forecasting engine is obtained as follows:

\[
y = \sum_{i=1}^{n} w_i F_i(x_1, x_2, \ldots, x_m) + \sum_{j=1}^{m} y_j
\]

(3)

In Error! Reference source not found., \( w_i \) indicates weights between the \( i_{th} \) neuron and the output, while the \( j_{th} \) input and the output are connected by \( V_j \), respectively. The parameters of WNNs, which should be specified by the improved clonal section as a training method, are shown as follows:

\[
S = \left[ y_1, \ldots, y_m, w_1, \ldots, w_n, a_1, \ldots, a_n, b_1, \ldots, b_n \right]
\]

(4)

2) Extreme Learning Machine-based Neural Network: ELM is a type of training strategy that is used in a single hidden-layer Feed-Forward Neural Network (FFNN) as shown in Fig. 4 [23], [25], [28].

In ELM, the input weights and biases are randomly selected, and only the output weights are determined mathematically using a simple matrix. Considering various samples, \( \{(x_i, t_i)\}_{i=1}^{M}, x_i \in \mathbb{A}^n \) where \( x_i = [x_{i1}, x_{i2}, \ldots, x_{in}] \) and \( t_i \in \mathbb{A}^n \) where \( t_i = [t_{i1}, t_{i2}, \ldots, t_{in}] \), the ELM having \( Y \) hidden neurons and the activation function \( q(\cdot) \) can estimate \( M \) samples without errors and can be formulated as follows:

\[
f(x_i, w, b, \beta) = \sum_{j=1}^{Y} \beta_j q(w_i x_j + b_j) = t_i, \quad j = 1, \ldots, M
\]

(5)

In (5), \( w_i \) is the weight between input data and hidden neurons, and \( \beta_j \) is the weight between hidden neurons and output nodes. \( b_j \) indicates the bias value of the hidden neurons. For the simplicity purpose, Eq. Error! Reference source not found. can be expressed as follows:
$H \beta = T$

$$H = \begin{bmatrix} q(w_1x_1 + b_1) & \cdots & q(w_ry_r + b_r) \\ \vdots & \ddots & \vdots \\ q(w_1x_M + b_1) & \cdots & q(w_ry_M + b_r) \end{bmatrix}_{M \times Y}$$

(6)

(7)

where $H$ indicates the matrix of the hidden nodes output. $\beta$ is a matrix connecting hidden nodes to output nodes. $T$ represents the target matrix. As in this strategy, the input weight and hidden neuron biases are produced randomly, the matrix of the hidden neuron output can be specified. Accordingly, the approximated output weights $\beta^*$ can be obtained by minimizing the following objective:

$$H \left( w_1^*, \ldots, w_r^*, b_1^*, \ldots, b_r^* \right) \beta^* - T = minH \left( w_1^*, \ldots, w_r^*, b_1^*, \ldots, b_r^* \right) \beta - T$$

(8)

The estimated output weight can be computed using the following simple inverse:

$$\beta^* = H^T H \beta$$

(9)

In (9), $H^T$ indicates the Moore-Penrose generalized inverse of the matrix associated with the output of hidden neurons $H$. Thanks to this simple matrix, the training speed is very high [28].

3) Multi-Layer Perceptron Neural Network: The MLP neural network is a FFNN model, which is a specific form of supervised NNs. MLP is capable of creating a mapping function between input datasets and the related output data. The MLP structure consists of multiple stacked layers of nodes, so-called neurons in neural networks. Each layer is connected to the next layer through the neurons in a one-directional manner. The general structure of the MLP neural network is shown in Fig. 5 [22], [29].

As shown in Fig. 5, the MLP structure consists of three layers: 1) Input layer, which feeds the network with input variables; 2) Output layer, which produces the final outputs; 3) Hidden layers, which are the stacked layers of neurons between input and output layers. The general process of the network is done through neurons, which consist of the activation function. Various activation functions can be used in the MLP structure, such as linear, logarithmic sigmoid, and hyperbolic tangent sigmoid functions. The MLP parameters that are determined through the training process include: weights connecting the input layer to a hidden layer; weights connecting a hidden layer’s output to the next hidden layer or the output layer; and hidden biases [22], [29].

Each layer can be represented mathematically as (10).

$$O_l^{(i)} = \Phi(a_l^{(i)}) = \Phi(\sum_{j=1}^{n_l} O_l^{(i-1)}w_{ij}^{(l)} + w_{0l}^{(l)}), \quad 1 \leq l \leq L$$

(10)

where $l$ represents the considered $L$ layer out of non-input layers of the network, $n_l$ shows the number of the neurons of the layer $l$, $O_l^{(i)}$ is the output of the neuron $i$ in the layer $l$, $w_{ij}^{(l)}, 1 \leq i \leq n_i$, denotes the weights related to the connection of the neuron $i$ in the layer $l$ with the neurons of the earlier layer $l - 1$, and $w_{0l}^{(l)}$ represents the bias of the neuron $i$ in the layer $l$. The first layer $l = 0$ is the input layer of the network, whose output length is $n_0$, and is characterized as $O^{(0)} = x$. Also, the last layer $l = L$ is the output layer of the network, whose output length is $n_L$ and is characterized as $O^{(L)} = y$. In (10), $\Phi$ is the activation function of the network, which is hyperbolic tangent sigmoid function in (11).

$$\Phi(x) = \tan sig(x)$$

(11)

4) Radial Basis Function Neural Network: The RBF neural network is another FFNN model. Its general performance is similar to the MLP neural network discussed previously. The general structure of RBF neural networks is shown in Fig. 6 [22], [30], which differs from its MLP counterpart in terms of the number of hidden neurons, only one hidden layer is used. In RBF neural networks, the activation functions of this structure are Radial Basis Functions. The mathematical equations of the RBF neural network are expressed by

$$Y = f(\sum_{j=1}^{n_l} Z_j v_j), \quad j = 1, 2, \ldots, n / \quad Z_j = f(\sum_{i=1}^{n_0} x_i w_{ij})$$

(12)

$$f(x) = e^{-\frac{a^2(x-b)^2}{2}}$$

(13)

where $v_j$ is the weight between the output and the hidden layer, $w_{ij}$ represents the weight between the input data and the hidden layer. $f()$ is the activation function for this neural network in this paper.
**B. TRAINING STRATEGY FOR WNN, RBF, AND MLP**

The ICSA is used to train WNN, RBF, and MLP models. The original Clonal Selection Algorithm (CSA) is firstly explained; the ICSA is then introduced [31].

- **Step 1**: The primary population of the clonal selection is randomly created in an acceptable range. This population, known as an antibody in the clonal selection, is a candidate solution for solving optimization problems. Each member of this population contains a parameter $\gamma$, known as a decision variable in the optimization problem, which can be determined from the optimal antibodies are created for the next generation.

- **Step 2**: The objective function is used as a criterion to specify the affinity of antibodies. In this problem, the training error, Mean Square Error (MSE), is considered as an objective function to be minimized.

- **Step 3**: According to the calculated objective function, the antibodies are sorted, and the antibody with the lowest value of the objective function is considered the best solution and ranked first.

- **Step 4**: The following equations can be used to replicate antibodies based on the position that they gain in Step 3.

\[
na_p = \text{Round}\left(\frac{\gamma N}{p}\right) \forall p = 1, 2, \ldots, N \tag{1}
\]

\[
NA = \sum_{p=1}^{N} \text{Round}\left(\frac{\gamma N}{p}\right) \tag{2}
\]

where $na_p$ is the number of antibodies replicated from $p_h$ antibodies. Round() is a function that rounds the obtained value to the nearest integer value. $\gamma$ is the rate of replication. $NA$ is the total number of replicated antibodies.

- **Step 5**: The total number of antibodies is mutated by

\[
S_{p,l} = \left[1 - e^{\frac{E_{\text{min}}}{E_{\text{max}}}}\right] \times S_{p,l} + e^{\frac{E_{\text{max}}}{E_{\text{min}}}} \left(S_{p,l} - S_{p,l}^{*}\right) \tag{3}
\]

1 $\leq p \neq p_i \neq p_j \leq NA, 1 \leq l \leq ND$

Eq. (16) is the first modification in the original CSA, where $S_{p,l}^{*}$ and $S_{p,l}$ demonstrate the $l_{th}$ gen of $p_h$ antibody in two consecutive generations. $E_{\text{min}}$ is the minimum MSE seen in the population. $ND$ is the number of gen in each antibody.

- **Step 6**: The objective function related to mutated antibodies is calculated in this step. The NT antibodies with the lowest MSE among the original antibodies are selected. This new set of antibodies go to the next generation.

- **Step 7**: In this step, $N - NT$ antibodies are created for the next generation.

- **Step 8**: While the number of generations is increased, this process ends when the termination criterion is met and reaches the maximum iteration of the optimization problem. Finally, the antibody with the lowest MSE is chosen as a solution. There is another modification added to the original CSA as shown in (17) based on which optimal antibodies are mutated less compared to others.

\[
N_p^M = \text{Round}\left(e^{\frac{E_{p}}{E_{\text{max}}}} \times ND\right) \tag{4}
\]

where $N_p^M$ indicates the number of gen associated with $p_{th}$ antibodies that are going to be mutated [22].

**IV. NUMERICAL ANALYSIS USING THE PROPOSED APPROACH**

In this section, a numerical analysis using the proposed technique for day-ahead wind speed forecasting using historical wind speed data measured in Saskatchewan is demonstrated. Two error evaluation indices are used to evaluate the performance of the proposed approach.

**A. DATA DESCRIPTION**

In this study, we use historical hourly wind speed data measured in Saskatoon, SK, Canada, which is obtained from Historical Climate Data on the Government of Canada website [32]. The rated power of a wind turbine of 1.8 MW is considered in this paper with the cut-in, rated, and cut-out wind speeds at 15 km/h, 50 km/h, and 90 km/h, respectively [33].

The measured wind speed, wind direction, humidity, and temperature are demonstrated in Fig. 7. The 60 days of these observed data before the prediction day are taken as the training data set. Accordingly, we have 1,440 hours of training data (24 hours × 60 days). March 2, 2021, December 1, 2020, September 1, 2020, and June 1, 2020 are chosen as the test days. Furthermore, the autocorrelation function is used to determine the most effective candidate input for the forecasting engine. More specifically, 400 hourly lagged values of wind speed, wind direction, humidity, and temperature are served as the candidate input data in this study, which are processed by the autocorrelation function to identify a minimum subset of the most informative features to be incorporated into the proposed model. The autocorrelation function results of wind speed for the test day on March 2, 2021 is shown in Fig. 8, which demonstrates the relationship between the input data and the target value. The data with a high autocorrelation function value is chosen as an input of the forecasting engine.
B. WIND SPEED FORECASTING EVALUATION

Each method should be evaluated in terms of performance and efficacy. For the performance evaluation, two factors arise: 1) data size, and 2) statistical criteria. The required data size depends on the employed method [34]. Statistical criteria are determined considering the nature of the methods. Therefore, it is necessary to review existing approaches for the performance evaluation in the context of wind speed forecasting. Since wind speed forecasting belongs to numeric methods, their performance evaluation is mostly based on numerical-error evaluations. In this regard, various criteria are used to evaluate the performance of wind speed forecasting, including MSE, Mean Bias Error (MBE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), and Skill Score [3], [4], [35], [36].

In this study, two error indices are used: the normalized Root Mean Square Error (nRMSE) and the normalized Mean Absolute Error (nMAE), which are formulated as follows:

\[
\text{nRMSE} = \frac{1}{NT} \sum_{t=1}^{NT} \left( \frac{WS_{\text{act}(t)} - WS_{\text{for}(t)}}{WS_{\text{max}}} \right)^2 \times 100
\]

\[
\text{nMAE} = \frac{1}{NT} \sum_{t=1}^{NT} \left( \frac{WS_{\text{act}(t)} - WS_{\text{for}(t)}}{WS_{\text{max}}} \right) \times 100
\]

where \(WS_{\text{act}(t)}\) and \(WS_{\text{for}(t)}\) are the measured and forecasted wind speeds at the time \(t\), respectively. \(NT\) is the number of hours on the test day. \(WS_{\text{max}}\) is the maximum wind speed on the test day.

C. CALCULATING WIND POWER FROM WIND SPEED

To obtain wind power using wind speed, the following equation is used:

\[
P(V) = \begin{cases} 
0, & 0 \leq V < V_{\text{cut-in}} \\
P_{\text{rated}} \times \frac{V - V_{\text{cut-in}}}{V_{\text{rated}} - V_{\text{cut-in}}}, & V_{\text{cut-in}} \leq V < V_{\text{rated}} \\
P_{\text{rated}}, & V_{\text{rated}} \leq V \leq V_{\text{cut-out}} \\
0, & V \geq V_{\text{cut-out}} 
\end{cases}
\]

D. INFLUENCE OF THE NUMBER OF FEATURES

In this study, five different hybrid NN techniques (WNN trained by ICSA, WNN trained by PSO, ELM, RBF, and MLP) are utilized to forecast Saskatchewan’s day-ahead wind speeds. To evaluate the impact of the number of features on the accuracy of wind speed prediction, two analyses have been performed: 1) Single-feature Analysis, only wind speed is considered as the input data, i.e., it only uses one feature; 2) Multi-feature Analysis, four input data are used including wind speed, wind direction, humidity, and temperature, i.e., it uses four features.

Single-feature Analysis: Table I shows the comparison of the performance of all five hybrid NN methods for the single-feature analysis using only wind speed data. The average
nRMSE for WNN trained by ISCA, ELM, RBF, MLP, WNN trained by PSO are 5.4059%, 6.9255%, 10.2936%, 12.4070%, and 17.0384%, respectively; the average nMAE for WNN trained by ISCA, ELM, RBF, MLP, and WNN trained by PSO, are 4.2893%, 5.4787%, 8.2527%, 9.5773%, and 13.4847%, respectively. The WNN trained by ICSA model has the lowest forecasting errors in all test days among all forecasting techniques comprising ELM, RBF, MLP, and WNN trained by PSO, and by 21.94%, 47.48%, 56.43%, and 68.27% of improvement for the average nRMSE; and by 21.71%, 48.02%, 55.21%, and 68.19% of improvements of the average nMAE, respectively. Therefore, the WNN trained by ICSA is the best forecasting engine. The ELM model also performs well.

On the other hand, ELM is a forecasting engine with high-speed and the least prediction time. In Table I, the ELM model forecasts wind speed in 4.76 seconds, the MLP, RBF, WNN trained by ICSA, and WNN trained by PSO models use 38.68 second, 523.83 seconds, 3.021 seconds, and 5.186 second, respectively. Such difference on computing time among models is due to different training strategies and different structures of neural networks. For instance, ICSA and PSO are iteration-based optimization methods, and they can discover the optimal solution after at least 100 iterations, which are time consuming training strategies. However, the ELM training approach is exceedingly fast due to a simple matrix computation using (7).

**Table I**

| Method | Error (%) | Test Day | Ave. Errors | Time (s) |
|--------|-----------|----------|-------------|----------|
| MLP    | nRMSE    | 10.0965  | 12.2553     | 15.7982  | 11.4779  | 12.4070  | 38.68    |
|        | nMAE     | 7.4607   | 10.2035     | 11.9463  | 9.1471   | 9.5773   |          |
| RBF    | nRMSE    | 6.1490   | 16.1394     | 7.0279   | 11.8581  | 10.2936  | 523.83   |
|        | nMAE     | 4.6678   | 13.7554     | 5.2165   | 9.5712   | 8.2527   |          |
| WNN (ICSA) | nRMSE | 4.1489   | 6.3779     | 5.1970   | 9.3799   | 5.4059   | 3.021    |
|        | nMAE     | 3.3086   | 5.3507     | 3.8404   | 4.8075   | 4.2893   |          |
| WNN (PSO) | nRMSE | 23.8552  | 12.2917     | 16.6545  | 15.3521  | 17.0384  | 5.186    |
|        | nMAE     | 19.1983  | 10.3228     | 12.9245  | 11.4931  | 13.4847  | 4.76     |

**Multi-feature Analysis:** The performance evaluation of forecasting engines considering four meteorological variables (wind speed, wind direction, humidity, and temperature) as features are shown in Table II. It is observed that WNN with the ICSA and ELM-based training strategies outperform other methods from the accuracy point of view. The ELM model also has the least computational time, which is significantly less than other models.

By comparing Tables I and II, we can see the multi-feature analysis for the day-ahead wind speed forecasting in this study does not improve accuracy of wind speed prediction for all models; in some cases, the accuracy is even reduced. It is mainly because when the number of input increases, it becomes difficult for the forecasting engine to find the best input/output mapping. Moreover, using multi-features lead to the increase of computational time for all models.

Therefore, in short-term wind speed prediction, the generated wind speed forecasts are merely based on the past wind speed data, and it is better not to include wind direction, temperature, and humidity in the forecasting model. However, these meteorological variables may be useful for a longer forecasting horizon.

**E. DAY-AHEAD WIND SPEED AND WIND POWER FORECASTING RESULTS**

Figs. 9-12 illustrate wind speed and wind power prediction outcomes using the five hybrid NN models for all test days (June 1, 2020, September 1, 2020, December 1, 2020, and March 2, 2021) considering only wind speed as the input data. According to these figures, the WNN trained by ICSA and ELM models can provide satisfactory trends and ramps following the actual wind speeds and wind power; while the other three models, WNN trained by PSO, MLP, and RBF, may not be able to follow the measured time series accurately. In wind power prediction figures, the wind turbine cannot generate power when the wind speed is lower than the cut-in wind speed or higher than the cut-out wind speed.

Scatter plots for wind speed prediction on September 1, 2020 are shown in Fig. 13 for all models. The scatter plots of WNN trained by PSO, MLP, and RBF models use 38.68 second, 523.83 seconds, and 5.186 second, respectively. Such difference on computing time among models is due to different training strategies and different structures of neural networks. For instance, ICSA and PSO are iteration-based optimization methods, and they can discover the optimal solution after at least 100 iterations, which are time consuming training strategies. However, the ELM training approach is exceedingly fast due to a simple matrix computation using (7).
FIGURE 9. The day-ahead wind speed forecasting results for June 1, 2020: (a) wind speed prediction, (b) wind power prediction.

FIGURE 10. The day-ahead wind speed forecasting results for September 1, 2020: (a) wind speed prediction, (b) wind power prediction.

FIGURE 11. The day-ahead wind speed forecasting results for December 1, 2020: (a) wind speed prediction, (b) wind power prediction.

FIGURE 12. The Day-ahead wind speed forecasting results for March 2, 2021: (a) wind speed prediction, (b) wind power prediction.
In this study, among the five evaluated hybrid NN methods, it is found that the WNN trained by ICSA model and the ELM model have better performance than other three methods. This is proved by both prediction accuracy and computation time aspects shown in Tables I and II. Figs. 9-12 graphically confirm that the WNN trained by ICSA and ELM models can accurately follow ramps and trends of the measured wind speed data. Finally, scatter plots affirm the superiority of the WNN trained by ICSA and ELM models over other methods.

Between the two best performing models, the ELM model is much more computationally efficient than the WNN trained by ICSA model, which can be valuable for conditions, in which the computation time plays a critical role. The WNN trained by ICSA model can be utilized in the planning stage since it has the highest forecasting accuracy.

V. CONCLUSION

Wind power is a promising clean energy source, especially in windy regions like Saskatchewan, Canada, but the intermittent nature of wind is one of the major challenges for wind power integration. In this paper, a novel hybrid neural network-based day-ahead wind speed forecasting technique is proposed with a comprehensive procedure. Five hybrid methods are considered using historical wind speed data recorded in Saskatchewan, Canada: 1) WNN trained by ICSA, 2) WNN trained by PSO, 3) ELM-based Neural Networks, 4) Radial Basis Function Neural Networks; and 5) Multi-Layer Perceptron Neural Networks. To evaluate the effect of the number of features, both single-feature analysis using only wind speed and multi-feature analysis using wind speed and other meteorological variables are conducted. It is found that for the day-ahead wind speed forecasting, only wind speed needs to serve as the feature, multi-features do not necessarily improve the accuracy, but increase the computation time.

Among the five hybrid NN models, the best performing models are the WNN trained by ICSA and ELM-based NN models. This is proven by the error indices calculation, graphically comparison between measured and forecasted wind speed curves, and scattered plots. Between the two best performing models, the WNN trained by ICSA model has the highest accuracy, while the ELM-based NN model has the least computation time. Therefore, which model to choose should be based on the particular applications.

The results in this study show that the proposed novel technique with a comprehensive procedure is very effective for day-ahead wind speed forecasting. The proposed method is very valuable to improve the economical, secure, and reliable operation of future wind power plants in Saskatchewan and beyond.

REFERENCES

[1] “The Path to 2030: SaskPower Updates Progress on Renewable Electricity.” https://www.saskpower.com/about-us/media-information/news-releases/2018/03/the-path-to-2030-saskpower-updates-progress-on-renewable-electricity (accessed Apr. 24, 2021).

[2] Y. Wang, Z. Zhou, A. Botterud, and K. Zhang, “Optimal wind power uncertainty intervals for electricity market operation,” IEEE Trans. Sustain. Energy, vol. 9, no. 1, pp. 199–210, 2017.

[3] H. S. Dhiman and D. Deb, “A Review of Wind Speed and Wind Power Forecasting Techniques,” ArXiv Prepr. ArXiv200902279, 2020.

[4] M. Santhosh, C. Venkaiah, and D. M. Vinod Kumar, “Current advances and approaches in wind speed and wind power forecasting
for improved renewable energy integration: A review,” Eng. Rep., vol. 2, no. 6, p. e21278, 2020.

[5] A. Shobana Devi, G. Maragatham, K. Boopathi, M. C. Lavanya, and R. Saranya, “Long-Term Wind Speed Forecasting—A Review,” in Artificial Intelligence Techniques for Advanced Computing Applications. Singapore, 2021, pp. 79–99.

[6] J. Dobchinski et al., “Uncertainty forecasting in a nutshell: Prediction models designed to prevent significant errors,” IEEE Power Energy Mag., vol. 15, no. 6, pp. 40–49, 2017.

[7] O. Karakuş, E. E. Kuruoğlu, and M. A. Altınkaya, “One-day ahead wind speed/power prediction based on polynomial autoregressive model,” IET Renew. Power Gener., vol. 11, no. 11, pp. 1430–1439, 2017.

[8] E. Yatıyana, S. rajakaruna, and A. Ghosh, “Wind speed and direction forecasting for wind power generation using ARMA model,” in 2017 Australasian Universities Power Engineering Conference (AUPEC), 2017, pp. 1–6.

[9] V. Radziukynas and A. Klementavicius, “Short-term wind speed forecasting with ARIMA model,” in 2014 55th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON), 2014, pp. 145–149.

[10] R. G. Kavasseri and K. Seetharaman, “Day-ahead wind speed forecasting using f-ARIMA models,” Renew. Energy, vol. 34, no. 5, pp. 1388–1393, 2009.

[11] C. Li and J.-W. Hu, “A new ARIMA-based neuro-fuzzy approach and swarm intelligence for time series forecasting,” Eng. Appl. Artif. Intell., vol. 25, no. 2, pp. 295–308, 2012.

[12] S. M. Verma, V. Reddy, K. Verma, and R. Kumar, “Markov models based short term forecasting of wind speed for estimating day-ahead wind power,” in 2018 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS), 2018, pp. 31–35.

[13] N. Amjad, F. Keynia, and H. Zareipour, “A new hybrid iterative method for short-term wind speed forecasting,” Eur. Trans. Electr. Power, vol. 21, no. 1, pp. 581–595, 2011.

[14] M. Khodayar, O. Kaynak, and M. E. Khodayar, “Rough deep neural network and Elman neural network,” Energy Convers. Manag., vol. 157, pp. 1–12, 2018.

[15] M. A. Mohandes, T. O. Halawani, S. Rehman, and A. A. Hussain, “Support vector machines for wind speed prediction,” Renew. Energy, vol. 29, no. 6, pp. 939–947, 2004.

[16] H. Malik, “Application of artificial neural network for long term wind speed prediction,” in 2016 Conference on Advances in Signal Processing (CASP), 2016, pp. 217–222.

[17] K. R. Nair, V. Vanitha, and M. Jisma, “Forecasting of wind speed using ANN, ARIMA and Hybrid models,” in 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICIICoIT), 2017, pp. 170–175.

[18] E. Calenas and W. Rivera, “Wind speed forecasting in three different regions of Mexico, using a hybrid ARIMA–ANN model,” Renew. Energy, vol. 35, no. 12, pp. 2732–2738, 2010.

[19] S. Osama, E. H. Houssein, A. E. Hassnain, and A. A. Fahmy, “Forecast of wind speed based on whale optimization algorithm,” in Proceedings of the 1st International Conference on Internet of Things and Machine Learning, New York, NY, USA, Oct. 2017, pp. 1–9.

[20] A. Aghjani, R. Kazemzadeh, and A. Ebrahimi, “A novel hybrid approach for predicting wind farm power production based on wavelet transform, hybrid neural networks and imperialist competitive algorithm,” Energy Convers. Manag., vol. 121, pp. 232–240, 2016.

[21] H. Chitsaz, N. Amjad, and H. Zareipour, “Wind power forecast using wavelet neural network trained by improved Clonal selection algorithm,” Energy Convers. Manag., vol. 89, pp. 588–598, 2015.

[22] C. Wan, Z. Xu, Y. Wang, Z. Y. Dong, and K. P. Wong, “A Hybrid Approach for Probabilistic Forecasting of Electricity Price,” IEEE Trans. Smart Grid, vol. 5, no. 1, pp. 463–470, Jan. 2014.

[23] H. Liu, X. Mi, and Y. Li, “Wind speed forecasting method based on deep learning strategy using empirical wavelet transform, long short term memory neural network and Elman neural network,” Energy Convers. Manag., vol. 156, pp. 498–514, Jan. 2018.

[24] C. Yu, Y. Li, and M. Zhang, “An improved Wavelet Transform using Singular Spectrum Analysis for wind speed forecasting based on Elman Neural Network,” Energy Convers. Manag., vol. 148, pp. 895–904, Sep. 2017.

[25] H. Jianjun, Y. Chuanjin, L. Yongle, and X. Huoyue, “Ultra-short term wind prediction with wavelet transform, deep belief network and ensemble learning,” Energy Convers. Manag., vol. 205, p. 112418, Feb. 2020.

[26] M. Santosh, C. Venkiah, and D. M. Vinod Kumar, “Ensemble empirical mode decomposition based adaptive wavelet neural network method for wind speed prediction,” Energy Convers. Manag., vol. 168, pp. 482–493, Jul. 2018.

[27] C. Wan, Z. Xu, P. Pinson, Z. Y. Dong, and K. P. Wong, “Probabilistic Forecasting of Wind Power Generation Using Extreme Learning Machine,” IEEE Trans. Power Syst., vol. 29, no. 3, pp. 1033–1044, May 2014.

[28] A. A. Ewees, M. A. Elaziz, Z. Alameer, H. Ye, and Z. Jianhua, “Improving multi-layer perceptron neural network using chaotic grasshopper optimization algorithm to forecast iron ore price volatility,” Resour. Policy, vol. 65, p. 101555, Mar. 2020.

[29] M. Madhiarasans and S. N. Deepa, “Comparative analysis on hidden neurons estimation in multi layer perceptron neural networks for wind speed forecasting,” Artif. Intell. Rev., vol. 48, no. 4, pp. 449–471, Dec. 2017.

[30] L. N. de Castro and F. J. V. Zuben, “Learning and optimization using the clonal selection principle,” IEEE Trans. Evol. Comput., vol. 6, no. 3, pp. 239–251, Jun. 2002.

[31] E. and C. C. Canada, “Historical Data - Climate - Environment and Climate Change Canada,” Oct. 31, 2011. https://climate.weather.gc.ca/historical_data/search_historic_data_e.c_h (accessed Apr. 26, 2021).

[32] “Centennial Wind Power Facility,” https://www.saskpower.com/our-power-future/our-electricity/electricity-system/system-map/centennial-wind-power-facility (accessed Apr. 24, 2021).

[33] J. Hännikäinen, “Multi-step forecasting in the presence of breaks,” J. Forecast., vol. 37, no. 1, pp. 102–118, 2018.

[34] M. Abbasipour, M. A. Igder, and X. Liang, “Data-Driven Wind Speed Forecasting Techniques Using Hybrid Neural Network Methods,” in 2021 Canadian Conference of Electrical and Computer Engineering (CCECE), 2021, pp. 1–6.

[35] H. Shaker, H. Zareipour, and D. Wood, “On error measures in wind forecasting evaluations,” in 2013 26th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), 2013, pp. 1–6.

Mehdi Abbasipour was born in Hamedan in Iran on February 3, 1994. He received his B.Sc. degree in Electrical Engineering (minor in Power Systems) from Shahid Beheshti University (National University of Iran, Sabzevar (USask), Saskatoon, SK, Canada. He has authored or co-authored several journal and conference papers. His research interests include Power Electronics and Power Systems, HVDC Grids and Systems, FACTS and DC Power Flow Controllers, Converters and Inverters, Machine Learning, and Renewable Energy Sources. He serves as a reviewer for IEEE Transaction on Power Systems and the IET journal on High Voltage.
Mosayeb Afshari Igder received the M.S degree in electrical engineering from Shiraz University of Technology, Shiraz, Iran, in 2016. He had a research collaboration with the Department of Electrical and Electronics Engineering, Shiraz University of Technology, from 2016 to 2020. He is currently pursuing his second Master of Science with University of Saskatchewan, Saskatoon, Canada. His research interests include power system operation, renewable energy, and marine power systems.

Xiaodong Liang (M’06–SM’09) was born in Lingyuan, Liaoning, China. She received the B.Eng. and M.Eng. degrees from Shenyang Polytechnic University, Shenyang, China in 1992 and 1995, respectively, the M.Sc. degree from the University of Saskatchewan, Saskatoon, Canada in 2004, and the Ph.D. degree from the University of Alberta, Edmonton, Canada in 2013, all in electrical engineering.

From 1995 to 1999, she served as a lecturer at Northeastern University, Shenyang, China. In October 2001, she joined Schlumberger in Edmonton, Canada, and was promoted to be a Principal Power Systems Engineer with this world’s leading oil field service company in 2009. After serving Schlumberger for almost 12 years, from 2013 to 2019, she was with Washington State University in Vancouver, WA, United States and Memorial University of Newfoundland in St. John’s, NL, Canada as an Assistant Professor and later an Associate Professor. In July 2019, she joined the University of Saskatchewan in Saskatoon, Canada, where she is currently an Associate Professor and Canada Research Chair in Technology Solutions for Energy Security in Remote, Northern, and Indigenous Communities. Her research interests include power systems, renewable energy, and electric machines.

Dr. Liang is a registered Professional Engineer in the province of Saskatchewan, Canada.