21th International Conference on Renewable Energies and Power Quality (ICREPQ'23)
Madrid (Spain), 24th to 26th May 2023
Renewable Energy and Power Quality Journal (RE&PQJ)
ISSN 2172-038 X, Volume No.21, July 2023

Very Short-Term Wind Power Forecasting Using a Hybrid LSTM-Markov Model Based on Corrected Wind Speed

A. Nuttapat Jittratorn [1], B. Chao-Ming Huang [2] and C. Hong-Tzer Yang [1]

[1] Department of Electrical Engineering
National Cheng Kung University
Tainan, Taiwan 701
e-mail: n28107014@gs.ncku.edu.tw, htyang@mail.ncku.edu.tw

[2] Department of Electrical Engineering
Kun Shan University
Tainan, Taiwan 710
e-mail: cmhuang@mail.ksu.edu.tw

Abstract. A Markov chain (MC) model is a statistical method of predicting future outcomes using past experience. This study proposes a hybrid method that uses a long short-term memory (LSTM) and a MC method to produce very accurate short-term (10-min) forecasts for the power output from a wind turbine (WT). The proposed method has three stages. The first stage uses k-means clustering to partition the wind power data into several clusters. The second stage uses LSTM models to initially predict the wind power output for each cluster. The final stage uses a MC method to construct the transition probability matrix for every 10-mimute time period. Using the transition probability matrices, the final predicted value for the WT power output is estimated using the prediction results for each cluster in the LSTM. This article also suggests a wind speed correction approach to enhance the forecasted wind speed result achieved by applying the weather research and forecasting model in order to generate more accurate wind power forecasting results. The proposed method is tested using a 3.6 MW WT power generation system that is located in Changhua, Taiwan. The effectiveness of the proposed model is compared with support vector regression (SVR), random forest (RF), LSTM and bidirectional gated recurrent unit (Bi-GRU) methods.

Key words. Wind turbine (WT) power generation, Markov chain, long short-term memory, wind speed correction.

1. Introduction

Wind power is a vital part of the global electricity supply because it features low environmental emissions and low power generation costs. The limited predictability of intermittent wind power generation means that system operation and market trading are dependent on the quality of wind power forecasting (WPF) [1]. Predictions of wind power are critical to the reliable and cost-effective operation of wind power systems, especially for a smart grid with distributed and highly linked generation.

For the planning of a renewable energy power generation and usage, especially in grid systems, accurate short-term wind speed forecasts are essential. In meteorology, it is common practice to use post-processing techniques based on outputs from weather prediction models and local data to enhance forecasts [2].

Wind power forecasting models are classified as persistent, physical, statistical, artificial intelligence/ machine learning (AI/ML) or hybrid models [3]. The persistence model treats the wind power for the immediate future the same as the current wind power. Physical modelling methods create mathematical models to determine the meteorological evolution of wind speed and then make wind power forecasts using the physical link between wind power conversion and meteorological conditions [4]-[5]. Probabilistic forecasting techniques [6]-[8] derive the probability distribution for wind power at the prediction stage and this is used to determine the risk for power system, dynamic dispatch, and unit commitment. This method involves quick processing and high precision qualities so the MC model [9]-[10] is extensively used for short-term and very short-term wind speed (or wind power) probabilistic forecasting among the probabilistic forecasting approaches.

MC prediction [11]-[12] forecasts the occurrence rate for future events based on the current stage using transfer probability, which shows the degree of the effect of many types of random factors. It is used for stochastically volatile prediction problems, particularly in the context of wind power. The MC method is used to predict wind power points and a weighted MC [13] improves data mining of the initial data to increase accuracy.

The focus of this study is using a MC model for very short-term wind power forecasting. The MC method is used with a LSTM in a hybrid model. To verify the performance of the proposed hybrid method, the SVR, RF, LSTM and Bi-GRU models are used. All models use previous point power outputs from a wind power plant, and correct wind speed and wind direction from weather research and forecasting model with a 10-minute resolution. The wind power prediction models are compared using performance measures, such as mean relative error (MRE), root mean square error (RMSE) and normalized root mean square error (NRMSE). The accuracy of a prediction model is determined by comparing the predicted and actual wind power outputs.

This paper is organized as follows. Section II gives an overview of the theory of clustering algorithms and single prediction models is given. The proposed method is described in Section III. Section IV presents the simulation results of wind speed correction and wind power forecasting for the single models and the proposed hybrid method. Conclusions are given in Section V.

2. Clustering and Single Prediction Models

A. The Unsupervised K-means Clustering Algorithm

One of the most basic and commonly used unsupervised machine learning techniques is k-means clustering. The k-means algorithm [14] is a clustering technique that uses Euclidean distance as the measurement index of similarity: the smaller the distance between the two samples, the greater is the similarity. The square error criterion function is stable at the minimum. The Euclidean distance and square error criterion function are defined as:

$$D(X,Y)=\{\sum_i \left|x_i-y_i\right|^2\}^{0.5} \tag{1}$$

$$J_C=\sum_{j=1}^{k}\sum_{i=1}^{n_i}\left\|x_i-m_j\right\|^2 \tag{2}$$

where k is the number of clusters, n_i is the number of samples in class J and m_j is the average value for the samples in class J. The k-means technique for data mining uses the first group of randomly picked centroids as the starting points for each cluster and then performs iterative (repetitive) computations to optimize the position of the centroids.

B. Random Forest

A machine-learning method called random forest is applied to classification and regression issues. An assortment of decision trees makes up a random forest. The bulk of the trees used for classification problems select the random forest output as their class, however the mean or average forecast for each individual tree is used for regression problems.

Bootstrap aggregation, commonly known as the ensemble technique of bagging, is used by random forest. Rows are picked to build each model using the bootstrap samples from the original data since bagging randomly selects a sample from the original dataset. Row sampling with replacement employs the bootstrap technique. Each individually trained model is used to produce the findings. When all the models are integrated, a majority vote or mean decision is reached. Aggregation is the process of merging all of the outputs to produce output. The RF model is less sensitive to noise and resilient to missing values and outliers. A thorough model of RF is presented in [15].

C. The Ensemble Learners

The forecasting outcomes of five different RF for wind speed correction are combined in this study using the linear regression (LR) and SVR models. A support vector machine (SVM) uses a maximum margin hyperplane to divide data points into two classes. This hyperplane is at the greatest distance between the nearest data points from two different classes, as shown in Figure 1. The support vectors are the data points on the margin. An SVM is used for classification and regression problems. Support vector regression (SVR) [16] constructs a dependency between the input and output of a system using a collection of training samples. The relationship is then used to anticipate the system output using the input.

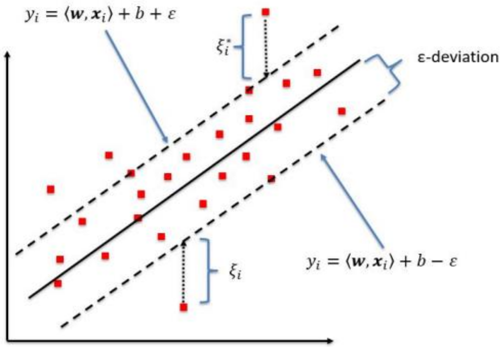


Fig. 1 Schematic diagram of the SVR model.

The two dashed lines represent the limits, which are a distance ε away from the reference data, where ε is a user-defined value. To develop the model, SVR only uses values outside the dotted lines. Training the SVR involves solving the following problems:

$$\text{minimize } \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\zeta_i^* + \zeta_i) \tag{3}$$

$$\textit{subject to}:\ \mathrm{y}_i - \langle w, x_i \rangle - b \le \varepsilon + \zeta_i^* \tag{4}$$

$$\textit{and}:\ \langle w, x_i \rangle + b - \mathrm{y}_i \le \varepsilon + \zeta_i \tag{5}$$

where w represents the learned weight vector, x_i represents the i-th training instance, y_i is the training label and $\boldsymbol{\zeta_i}$ represents the gap between the boundaries and projected values beyond the bounds. C is a constraint that governs the penalty that is applied to observations that occur outside the boundaries.

D. Long Short-Term Memory (LSTM)

The LSTM recurrent neural network has a self-loop network structure to memorize the previous information and apply it for calculating the current output [17]. Because of its structure, LSTM can avoid long-term dependency problems. Figure 2 illustrates the data flow and controlling through a memory cells and gates.

The calculation formulae among variables are shown in (6) to (11). The first step is to decide what information to discard through the forgotten gate by equation (6).

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{6}$$

The next step is to decide how much new information to add to the cell state by (7) and (8).

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{7}$$

$$O_t = \sigma(W_O \cdot [h_{t-1}, x_t] + b_o) \tag{8}$$

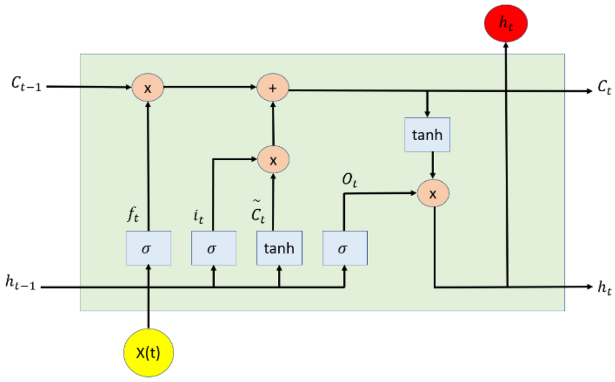


Fig. 2 The internal structure of a LSTM unit.

where W_f, W_t and W_0 are weight matrix for each gate and b_f, b_t and b_0 are the bias for each gate. Then, the cell state can be updated by (9),

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \quad (9)$$

Lastly, determine what values to output by (10) and (11). This output will be based on cell state.

$$\tilde{C}_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (10)$$

$$h_t = O_t \cdot \tanh(C_t) \quad (11)$$

where W_C and b_C are the weight matrix and the bias for each gate, respectively.

D. Markov Chain

Using specific assumptions and probabilistic criteria, Markov models are characterized as stochastic processes with random variables that perform transition from one state to the next state. Therefore, the chance of a random process transitioning to the next feasible state is determined by the present state and is not significantly affected by prior states.

To construct the transition probability matrix, $\{\boldsymbol{x_t}\}_{t\geq 0}$ is a sequence of discrete random numbers. If the sequence $\{\boldsymbol{x_t}\}_{t\geq 0}$ obeys the following equality, it is a MC and is expressed as [18]:

$$\begin{aligned} &P\{X_{t+1} = j / X_t = i, X_{t-1} = i_{t-1}, ..., X_0 = i_0\} \\ &= P\{X_{t+1} = j / X_t = i\} = p_{ij} \end{aligned} \quad (12)$$

where $t = 1, 2, 3, \ldots$ for all states i and j.

As shown in (12), only the most recent data in the series determines what happens in the next state. The equation shows that a MC is homogeneous because transition probabilities are independent of time-shifts. The conditional probabilities are mapped in a transition matrix (P). For n possible states, the transition matrix is expressed as

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{bmatrix} \quad (13)$$

where $p_{ij} = \frac{a_{ij}}{\sum_{i=1}^{n} a_{ij}}$ for all states i and j. All the elements of P are non-negative ($0 \leq p_{ij} \leq 1$) and the sum of every row is $\sum_{j=1}^{n} p_{ij} = 1$. a_{ij} is the number of transitions from states i to j. A transition probability matrix P is shown in Figure 3.

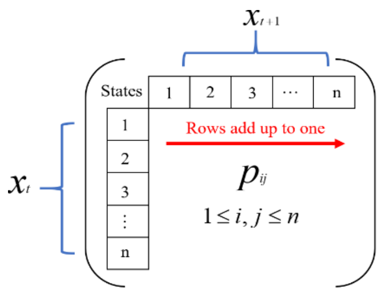


Fig. 3 The properties of a transition probability matrix.

E. Weighted Markov Chain

A weighted Markov chain [19] is used to determine the effect of each step's transition probability matrix and for data mining. A weighted MC involves the following stages:

(1) Develop categorization criteria and apply these to each data set to determine its status.
(2) At each step, create the Markov chain transition probability matrix.
(3) Starting with the preceding data, predict the state probability using the related transition probability matrix.
(4) For the Markov chain model, determine the weight of each step.
(5) The final probability is the weighted sum of each predictor's probability for the same state.
(6) Steps 1 through 5 are repeated for the subsequent round of prediction, adding the predicted value to the original data series.

3. The Proposed Methods

A. Wind Speed Correction

The proposed wind speed ensemble forecasting system uses the clustering method, classification strategies, RF models, and the regression-based ensemble model. The general structure of ensemble wind speed forecast is depicted in Figure 4. Three processes make up the general framework: model training, creating an ideal set of weights, and model testing. The historical wind speed for k different weather situations is used in step 1 of the k-means clustering algorithm. Label as breeze, moderate, cool, strong, and powerful wind are the five clusters. Then, using the historical and forecasted weather 10-minute data as input and wind speed as an output, an RF model was built on each cluster. Additionally, the historical wind speed data used as input and the label produced by k-means clustering as output were used to train a k-nearest neighbour (KNN) classification model. The model determines the effect of the historical wind speed at time t-1 and the forecasted wind speed and wind direction at time t from weather research forecasting in order to correct the wind speed at time t.

B. Wind Power Forecasting

The proposed method uses k-means clustering to classify historical WT power data based on power distribution by constructing several LSTM prediction models and establishing a weighted Markov chain probability matrix. Figure 5 shows a schematic diagram of the proposed method. A Markov chain prediction uses a Markov chain model with a step size of 1 and an initial state

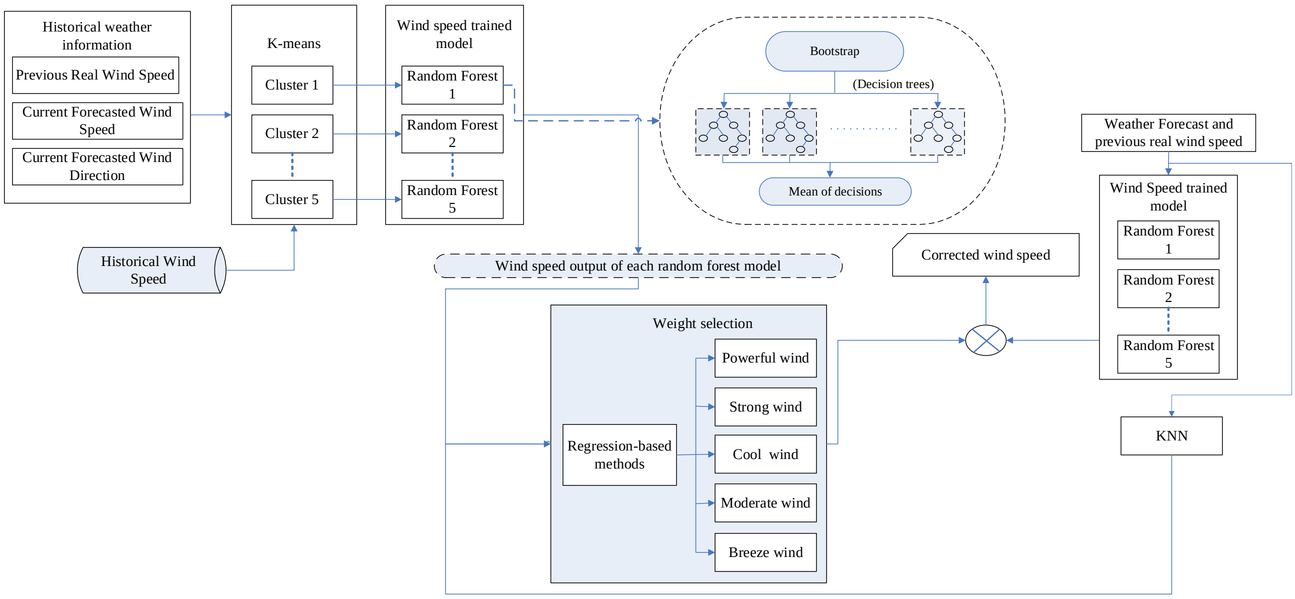


Fig. 4 Details of the wind speed correction model.

vector to determine the absolute distribution for a future time. The period between data collection points is 10 minutes. A total of 144 steps are required for each day. This study determines the effect of the historical power at time t-1, corrected wind speed and forecasted wind direction at time t, in order to forecast the power at time t. To determine the one-step ahead prediction value, the weighted Markov chain approach and LSTM models are used to calculate the final predicted value for the power by determining the weights at various phases.

Details of the proposed scheme are described as follows:

(1) The original data must be cleaned so missing values and outliers in the data must be processed. This study uses interpolation to supplement the missing values, which are then subject to the clustering process.

(2) The data is adjusted after clustering. The normalization method for this study is shown in (14). The range of data after normalization is 0 to 1:

$$x_{nor} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{14}$$

where x_{max} and x_{min} are the maximum and minimum values of the measured WT power data, respectively.

(3) The best LSTM model for each cluster (k_1, k_2, k_3, …, k_n) is trained separately. The predicted LSTM series is denoted as $(\hat{P}_{t+1}^{(k1)}, \hat{P}_{t+1}^{(k2)}, \hat{P}_{t+1}^{(k3)}, \ldots, \hat{P}_{t+1}^{(kn)})$.

(4) Finally, a Markov chain model is used based on the transition probability matrix, and the forecasting results for the LSTM model for each cluster are weighted and summed.

4. Numerical Results

The proposed method is tested using a 3.6 MW wind power generation system. The data was collected from January 2020 to December 2020 and includes wind power generation, wind speed and direction at a resolution of 10 minutes. Weather prediction data were collected from the Solcast.com platform, which is a weather research forecasting platform. The results for one-step ahead wind power forecasting using each single model are compared

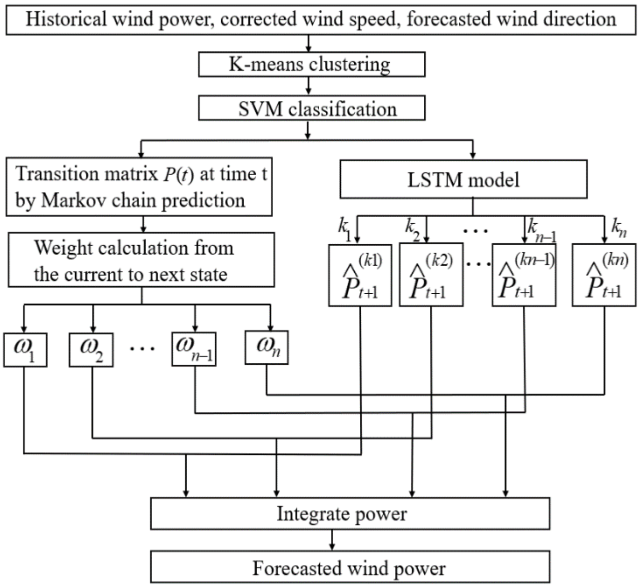


Fig. 5 Schematic diagram of the proposed method

for the wind power and speed data for the Changbin industrial area, Changhua, Taiwan. The proposed LSTM-Markov models are then used for this case study. All learning algorithms are coded entirely in Python 3.7, which uses the Keras API in conjunction with the TensorFlow framework. The training data covers the first three weeks of every month. The testing data contains last week's data for every month, which is randomly selected from a day in each month and a day of each season. There is four times more training data than validation data. The final wind power time series is shown in Figure 6.

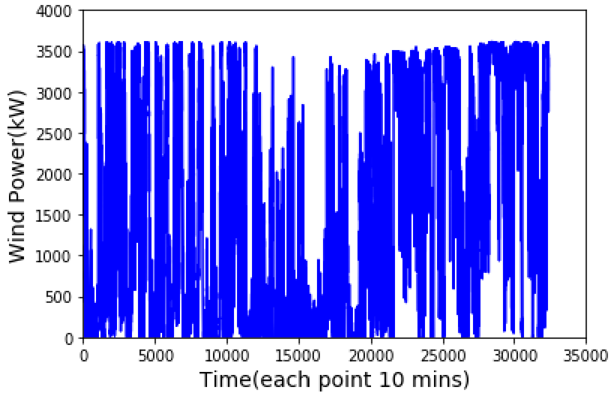
The test results for each model are discussed. To determine the accuracy of the forecast, MRE, RMSE and NRMSE are used:

$$\mathbf{MRE} = \frac{1}{N}\sum_{i=1}^{N} \frac{|P_{fore} - P_{true}|}{P_{cap}} \times 100\% \tag{15}$$

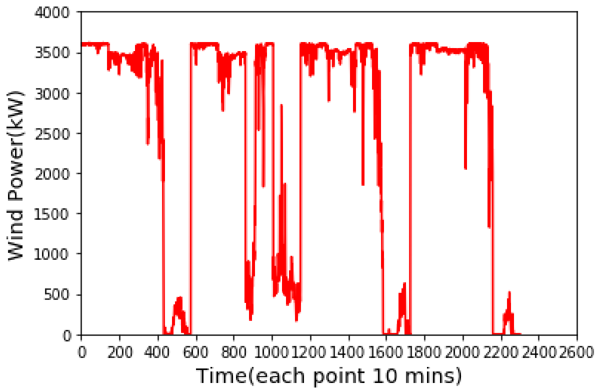
$$\mathbf{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(P_{fore} - P_{true}\right)^2} \tag{16}$$

$$\mathbf{NRMSE_{cap}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{P_{fore}-P_{true}}{P_{cap}}\right)^2} \times \mathbf{100\%} \quad (17)$$

where P_{fore} is the estimated value, P_{true} is the actual value, P_{cap} is the capacity for wind power generation and the maximum wind speed for wind speed measurement and N is the number of data points.



(a)



(b)

Fig. 6 The curves for (a) the training and (b) the testing datasets.

For this study, testing data is used, and performance is compared in terms of the forecasting results for the hybrid models and all single models as a power forecasting scenario. In the wind speed correction scenario, the forecasted wind speed from weather research forecasting and the corrected wind speed are compared.

Figure 7 shows the comparison results between forecasted wind speed and the corrected wind speed, at a resolution of 10 minutes. Table I lists the performance metrics for the forecasted wind speed and corrected wind speed results. The results show that the MRE, RMSE and NRMSE for the corrected wind speed are much better than the pure forecasted wind speed.

Figure 8 shows the one-step ahead forecasting results for four models and the results for the hybrid model, at a resolution of 10 minutes. Table II lists the performance metrics for the forecasting results for testing data for the different single models and the hybrid model. The results show that the proposed model gives the most accurate results for this case.

The prediction results for the different models are significantly different. Using this testing data, RF performs the worst of all models. The results for each dataset show that if WT power output fluctuates rapidly, the SVR model predictions also fluctuates, mainly because it climbs or decreases too quickly and by too great an amplitude, so outcomes are poor. The prediction results for the Bi-GRU also fluctuate over a short period of time, but not as much as the SVR model, so it has medium performance. The LSTM's forecasting results are quite flat and they approximately represent the variations in WT power generation. If the WT power output fluctuates quickly, the LSTM's predictions and the accuracy of the time when winds are strong is also reduced. The hybrid models benefit from each model's strengths to compensate for the weaknesses of other models, so they give the best overall performance.

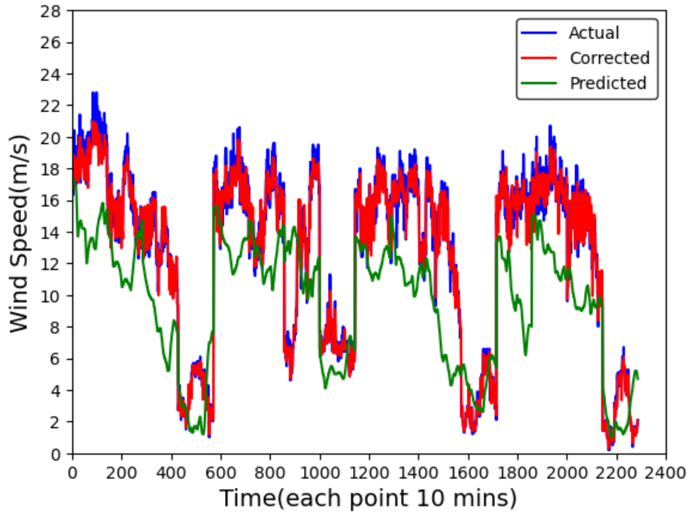


Fig. 7 One-step ahead wind speed correction results.

Table I. Forecasting Results for Wind Speed Forecasting and Correction

Model	MRE (%)	RMSE (m/s)	NRMSE (%)
Forecasted Wind Speed	16.88	4.56	20
Corrected Wind Speed	**2.61**	**0.82**	**3.58**

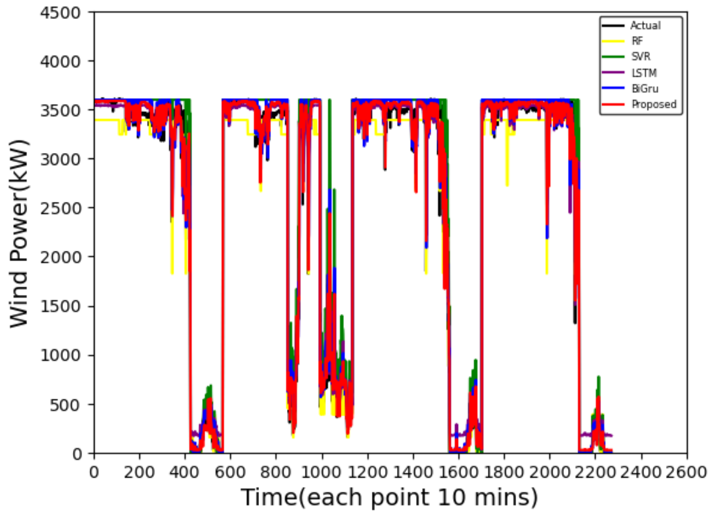


Fig. 8 One-step ahead WT power forecasting results for all models

Table II. Forecasting Results for Different Single Models and Proposed Hybrid Model

Model	MRE (%)	RMSE (kW)	NRMSE (%)
SVR	3.90	210.52	5.85
RF	4.27	271.22	7.55
LSTM	2.81	187.07	5.12
Bi-GRU	3.28	188.73	5.12
Proposed	**2.47**	**156.95**	**4.44**

5. Conclusions

This study uses a hybrid LSTM and Markov chain model for very short-term forecasting for WT power. The proposed method uses k-means clustering to partition wind power data into 5 clusters. LSTM models are then used to produce an initial prediction for each cluster. A weighted MC model and a LSTM are used to produce the final prediction. The results for testing on a 3.6 MW WT power generation system show that the RNN-based models such as LSTM and Bi-GRU give better accurate forecasts than all regression-based models such as RF and SVR; the LSTM performs respectably and the RF performs poorly. The hybrid models give a more accurate forecast than the single models because the hybrid model benefit from advantages of two or more models so prediction are more accurate. The proposed hybrid LSTM-MC model performs very well after wind speed correction processing, but there is scope for improvement for the proposed hybrid MC model and wind speed correction scenario. The proposed models could use a higher order of the Markov chain model and use more accurate corrected wind speeds to further increase forecasting accuracy.

Acknowledgement

The financial support of the National Science and Technology Council, Taiwan, R.O.C. under Grant No. 111-3116-F-006-005 is gratefully acknowledged.

References

[1] J. Yan, F. Li, Y. Liu and C. Gu, "Novel Cost Model for Balancing Wind Power Forecasting Uncertainty," *IEEE Trans. Energy Conversion*, vol. 32, no. 1, March. 2017, pp. 318-329.

[2] D. Choi, W. Shin, K. Ko and W. Rhee, "Static and Dynamic Yaw Misalignments of Wind Turbines and Machine Learning Based Correction Methods Using LiDAR Data," in IEEE Transactions on Sustainable Energy, vol. 10, no. 2, April 2019, pp. 971-982.

[3] M. U. Yousuf, I. Al-Bahadly and E. Avci, "Short-Term Wind Speed Forecasting Based on Hybrid MODWT-ARIMA-Markov Model," *IEEE Access*, vol. 9, 2021, pp. 79695-79711.

[4] Z. Sun and M. Zhao, "Short-Term Wind Power Forecasting Based on VMD Decomposition, ConvLSTM Networks and Error Analysis," *IEEE Access*, vol. 8, 2020, pp. 134422-134434.

[5] M. B. Ozkan and P. Karagoz, "Data Mining-Based Upscaling Approach for Regional Wind Power Forecasting: Regional Statistical Hybrid Wind Power Forecast Technique (RegionalSHWIP)," *IEEE Access*, vol. 7, 2019, pp. 171790-171800.

[6] C. Gilbert, J. Browell and D. McMillan, "Leveraging Turbine-Level Data for Improved Probabilistic Wind Power Forecasting," *IEEE Trans*. vol. 11, no. 3, July 2020, pp. 1152-1160.

[7] H. Zhang, Y. Liu, J. Yan, S. Han, L. Li and Q. Long, "Improved Deep Mixture Density Network for Regional Wind Power Probabilistic Forecasting," *IEEE Trans. Power Systems*, vol. 35, no. 4, July 2020, pp. 2549-2560.

[8] M. Cui, J. Zhang, Q. Wang, V. Krishnan and B. -M. Hodge, "A Data-Driven Methodology for Probabilistic Wind Power Ramp Forecasting," *IEEE Trans. Smart Grid*, vol. 10, no. 2, March. 2019, pp. 1326-1338.

[9] T. Hu, W. Wu, Q. Guo, H. Sun, L. Shi and X. Shen, "Very short-term spatial and temporal wind power forecasting: A deep learning approach," *CSEE Journal of Power and Energy Systems*, vol. 6, no. 2, June. 2020, pp. 434-443.

[10] M. J. Sanjari, H. B. Gooi and N. C. Nair, "Power Generation Forecast of Hybrid PV–Wind System," in *IEEE Trans. Sustainable Energy*, vol. 11, no. 2, April. 2020, pp. 703-712.

[11] L. Yang, M. He, J. Zhang and V. Vittal, "Support-Vector-Machine-Enhanced Markov Model for Short-Term Wind Power Forecast," in *IEEE Trans. Sustainable Energy*, vol. 6, no. 3, July. 2015, pp. 791-799.

[12] P. Jeyakumar, N. Kolambage, N. Geeganage, G. Amarasinghe and S. K. Abeygunawardane, "Short-term Wind Power Forecasting Using a Markov Model," *EECon*, 2021, pp. 19-24.

[13] X. Yang, X. Ma, N. Kang and M. Maihemuti, "Probability Interval Prediction of Wind Power Based on KDE Method with Rough Sets and Weighted Markov Chain," *IEEE Access*, vol. 6, 2018 pp. 51556-51565.

[14] K. P. Sinaga and M. Yang, "Unsupervised K-Means Clustering Algorithm," *IEEE Access*, vol. 8, 2020, pp. 80716-80727.

[15] Y. Xuan et al., "Multi-Model Fusion Short-Term Load Forecasting Based on Random Forest Feature Selection and Hybrid Neural Network," in IEEE Access, vol. 9, 2021, pp. 69002-69009.

[16] S. Zhang, M. Wang, P. Zheng, G. Qiao, F. Liu and L. Gan, "An Easy-to-Implement Hysteresis Model Identification Method Based on Support Vector Regression," *IEEE Trans. Magnetics*, vol. 53, no. 11, Nov. 2017, pp. 1-4.

[17] C. -H. Liu, J. -C. Gu and M. -T. Yang, "A Simplified LSTM Neural Networks for One Day-Ahead Solar Power Forecasting," in IEEE Access, vol. 9, 2021, pp. 17174-17195.

[18] C. Ren, J. Gu, S. Tian, J. Zhou, S. Shi and Y. Fu, "A Short-Term Rolling Prediction-Correction Method for Wind Power Output Based on LSTM and Markov Chain," *IEEE/IAS Industrial and Commercial Power System Asia (I&CPS Asia)*, 2021, pp. 574-580.

[19] H. Shi, Y. Li, M. Ding, X. Li, Z. Li and Z. Zhang, "Wind Power Prediction Based on Rough Set Conditional Entropy Weighted Markov Chain," *IEEE Sustainable Power and Energy Conference (iSPEC)*, 2021, pp. 169-172.