Predictive Analysis of Machine Learning Schemes in Forecasting Offshore Wind Speed

Predictive Analysis of Machine Learning Schemes in Forecasting Offshore Wind Speed

Mostafa Bakhoday-Paskyabi

Geophysical Institute, Bergen Offshore Wind Center, University of Bergen and Bjerknes Centre for Climate Research, Postbox 7803, 5020 Bergen, Norway

E-mail: Mostafa.Bakhoday-Paskyabi@uib.no

Abstract. High variability of wind in the farm areas causes a drastic instability in the energy markets. Therefore, precise forecast of wind speed plays a key role in the optimal prediction of offshore wind power. In this study, we apply two deep learning models, i.e. Long Short-Term Memory (LSTM) and Nonlinear Autoregressive EXogenous input (NARX), for predicting wind speed over long-range of dependencies. We use a four-month-long wind speed/direction, air temperature, and atmospheric pressure time series (all recorded at 10 m height) from a meteorological mast (Vigra station) in the close vicinity of the Havsul-I offshore area near Ålesund, Norway. While both predictive methods could efficiently predict the wind speed, the LSTM with update generally outperforms the NARX. The NARX suffers from vanishing gradient issue and its performance declines by abrupt variability inherited in the input data during training phase. It is observed that this sensitivity will significantly decrease by integrating, for example, the wind direction at low frequencies in the learning process. Generally, the results showed that the predictive models are robust and accurate in short-term and somewhat long-term forecasting of wind.

1. Introduction

Nowadays, wind energy is one of the most prospective sources of renewable energy due to global awareness on climate change, steady increase in the global population, and the environmental, political and economical issues related to the fossil fuels [1]. As it is essential maximizing production while reducing the structural loads to turbines and mitigating the wake losses (same as maximizing production), the high variability of wind in the farm areas can cause a drastic instability in the energy markets [2]. Therefore, precise forecast of wind speed and correspondingly the wind power generation plays a critical role on the optimal dispatch plans of grid control applications.

To tackle the challenges related to the wind speed forecasting, several studies have been conducted for forecasting of the wind speed in very short, short (i.e. 30 min to few hours), medium, and long term (i.e. more than a day). The proposed models are physical models [3], data-driven and statistical methods [4, 5], and hybrid models [7, 8]. The physical predictive models rely on governing equations of motions that have predictive ability across abroad range of spatial/temporal scales. However, these models are computationally expensive. Data-driven models rely on training of their weights based on past available train data and are more effective for short-term forecast, and hybrid models gain the performance by employing both models in there predictions. Comparative analyses have demonstrated that well-developed

data-driven/hybrid models may achieve better (or at least equivalent) prediction accuracy than physical models, see for example [9, 10].

Artificial Neural Network (ANN) with various configurations has been used for the short-term wind forecast, where forecasts beyond the last observations collected in the training data are only reliable for few time steps [6]. ANN (consisting of input layer, hidden layer and output layer) is called shallow if it uses only one hidden layer and deep if the number of hidden layers is more than one. Specifically, the internal structure of deep neural networks, such as Recurrent Neural Network (RNN), makes it possible to recover complicated nonlinearities in the data [11]. RNNs contain memory blocks with ability to remember information at each time from the previous samples. The Long Short-Term Memory (LSTM) is one of the most common kind of RNN for processing time series that can retain information for a long period of time during the learning process [12]. While ANN-based techniques are able to capture the nonlinear tendencies in the wind data, their performance is sensitive to the choice of reliable criteria (e.g. the network topology and the number of hidden neurons) which control the structure of the ANN. Nonlinear Autoregressive Networks with EXogenous input (NARX) is another kind of RNN in which multi-step forecast of wind speed can be performed by wind speed along with additional meteorological time series such as air temperature, atmospheric pressure, and wind direction [13].

In this study, we examine the predictability of wind at 10 m height using two different deep learning data-driven methodologies, i.e. LSTM and NARX, Section 3. In particular, we apply these methods to the wind time series of a meteorological mast in the Northwest coast of Norway between November 2011 and February 2012, Section 2. For the NARX approach in addition to the wind speed, we use the air temperature, pressure, and wind direction. Finally, we discuss in details the performance of models in forecasting wind by the means of their ability to capture the statistical properties of wind speed, i.e. Section 4.

2. Measurement site and data

The measurement site is in very close vicinity of the Havsul-I offshore area off the west coast of Norway (Fig. 1) which was the first site in Norway with a concession for construction of an offshore wind farm due to the wind potential of the region. Ancillary atmospheric data at 10 m height were logged from a meteorological station, i.e. Vigra station, with coordinates: 62.83°N and 6.15°E, Fig. 1.

Samples used in the predictive models correspond to an approximately five-month of hourly measurements of: wind speed (m/s), wind direction (°), air temperature (°C), and atmospheric pressure (bar) from October 15, 2011, to March 12, 2012, including 3577 pieces of data for each mentioned variable. Figure 2-a shows the wind speed time series used in this study. Wind is blowing on average from southeast and southwest, and wind speed in the range between 1 to 15 m/s contains several events, see more details in [14]. The maximum air temperature is about 15 °C for couple of days, i.e. Fig. 2-b. Moreover, high pressure fronts pass over the station during this period, Fig. 2-c.

.

3. Methods

The (point) forecast procedure for the wind speed u can be written as follows:

$$\begin{aligned} \hat{u}_{t+H|t} &= g(\text{current information}) && (1)\\ &= g(u_t, u_{t-1}, \cdots, u_{t-t_u}, X_t, X_{t-1}, \cdots, X_{t-t_X}), && (2) \end{aligned}$$

where $\hat{u}_{t+H|t}$ denotes the wind speed forecast at time t and H is the forecast horizon. g is either deterministic/stochastic or machine learning based functional map that makes a bridge between the current input information (e.g. wind speed and other atmospheric data) and the output

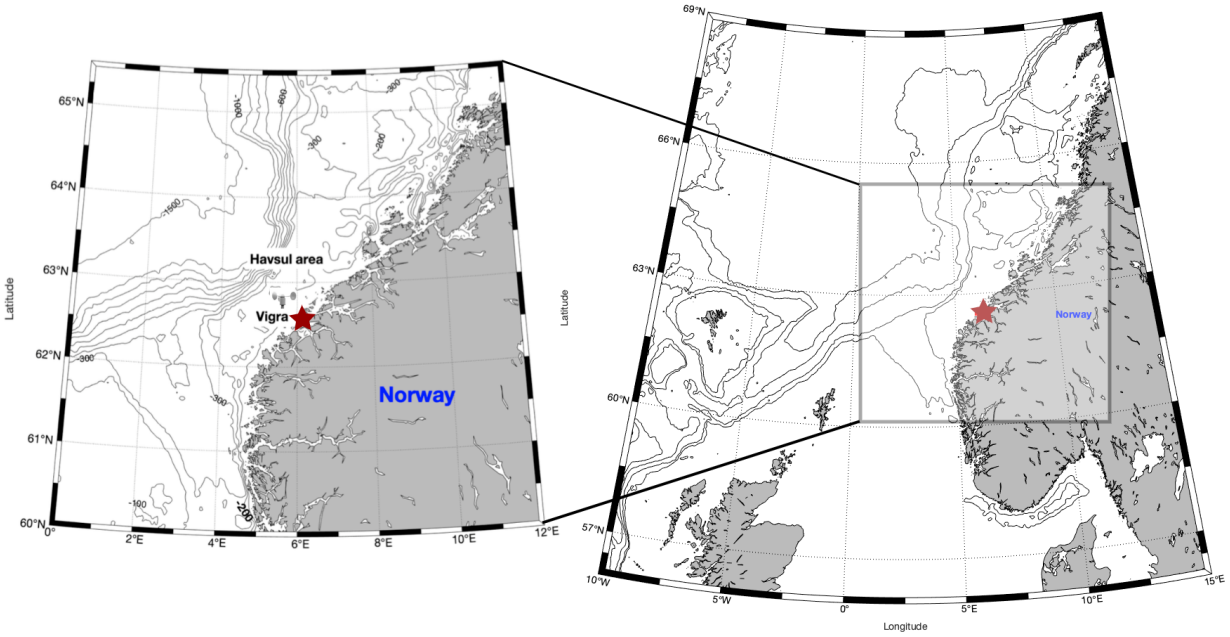

Figure 1. Location of the measurement site (i.e. meteorological station Vigra) close to the Havsul-I offshore area.

information corresponding to the future information. $\Omega = \{u_t, u_{t-1}, \cdots, u_{t-t_u}, X_t, \cdots, X_{t-t_X}\}$ is information set in which $X_t, X_{t-1}, \cdots, X_{t-t_X}$ are all other exogenous time series used in the forecast. t_u and t_X are values of first instant of input used in the learning process. As an example for $H = 1$, the forecast function g, in an univariate statistical form, can be simply defined by $\hat{u}_{t+1|t} = u_t$ for $\Omega = \{u_t\}$ or using rolling average with the same forecast horizon as $\hat{u}_{t+1|t} = \Sigma_{k=t-m}^{t} y_k/m$ for $\Omega = \{y_t, \cdots, y_{t-m}\}$ to utilize information from the last m samples. Typically in the real world, there is no analytical form to explain g due to all uncertainties/noise involved in the data measurements or modelling. Furthermore, these data may not meet some underlying assumptions required by the traditional statistical methods (e.g. stationarity requirement). Machine learning approaches have, however, introduced new paradigms to model forecast function g through a layered learning hierarchy which is not relying on time-ordering as it is essential for the statistical models [7]. While the statistical models in terms of processing are straightforward, the machine learning based approaches depend on different factors such as network structure design, model training, and hyper-parameter tuning, as well as quality of training historical/past information (i.e. their uncertainty and noise content). In this Section, we investigate both the LSTM and NARX methods due to their great learning ability to predict various time series.

3.1. LSTM

Feed forward neural networks process the relationships between each input and output independently. This may cause an issue so-called vanishing gradient (short-term memory) where the information from the previous learning is disappeared over the large time interval [15]. LSTM addresses the short-term memory issue of RNN by employing internal mechanisms to control flow of information to learn long-term dependencies. It generally contains 4 interacting layers:

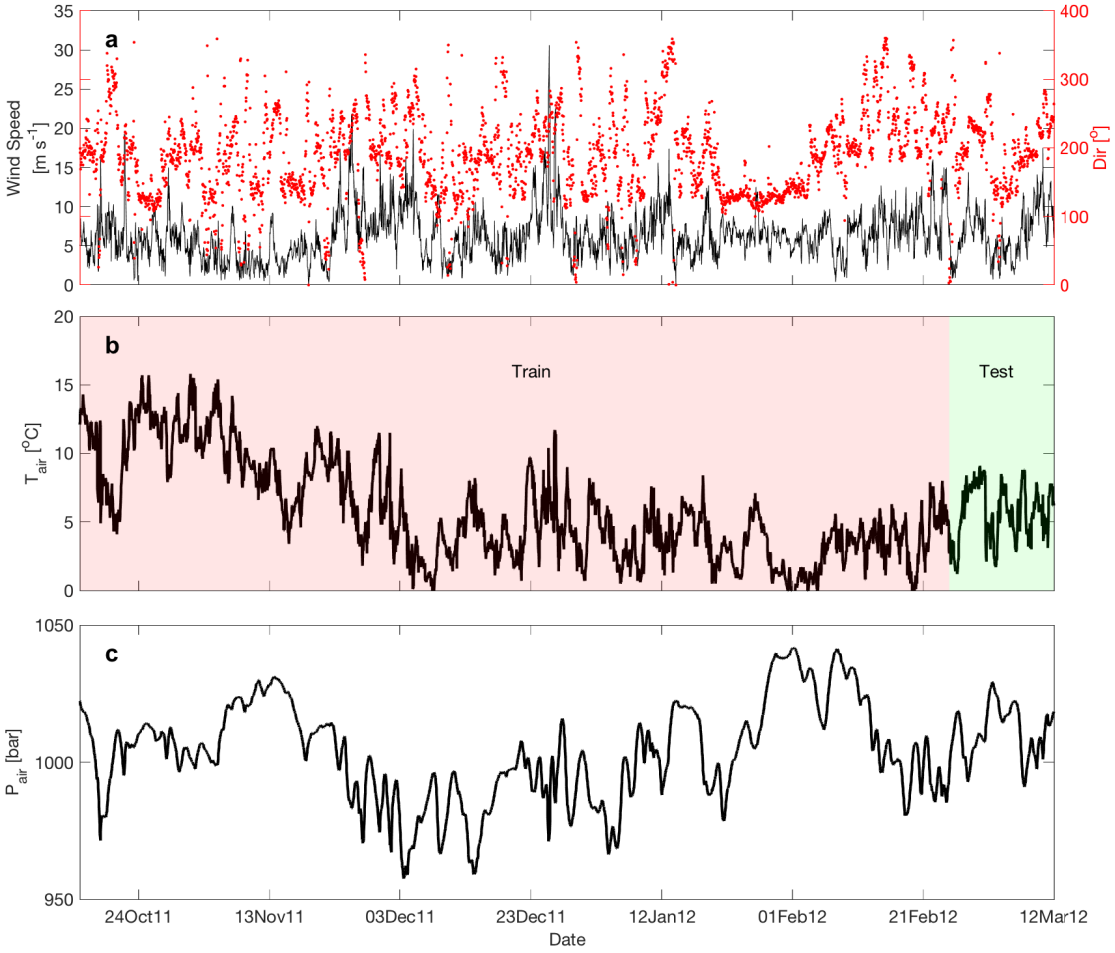

Figure 2. Time series of: (a) wind speed U_{10} (black) and wind direction (red markers) at 10 m height measured at Vigra station (see Fig. 1); (b) air temperature (black). Shaded areas in this plot are portions of data used for the training (red) and the test (green). They correspond to 90% and 10% of the total number of data, respectively; and (c) time series of atmospheric pressure.

input, hidden, output layers, and cell state (memory block) as a special structure for neurons. This architecture is simply a memory cell to remember information within the neurons in the hidden layer and three gates, Fig. 3-a. The cell state allows forward flow of data through applying a series of linear transformations, and add-to or removed-from the cell operators using sigmoid gates [12]. A gate, like a layer, contains individual weights.

As shown, the LSTM consists of an input gate, i_t, a forget gate, f_t, a cell gate c_t, and an output gate, o_t. This internal gate mechanism can solve the vanishing gradient problem in the training process by learning which data needs to be kept or discarded from the sequence. The input, output, and forget gates control the flow of data into and out of the cell. The general relationships for different gates are given as

$$X_t \quad = \quad \sigma(W_X \cdot [h_{t-1}, x_t] + b_X), \tag{3}$$

where $X = \{i, f, g, o\}$ contains the name of different gates, σ denotes the activation sigmoid function, and $[h_{t-1}, x_t]$ is the concatenated input signal. The sigmoid function is calculated by

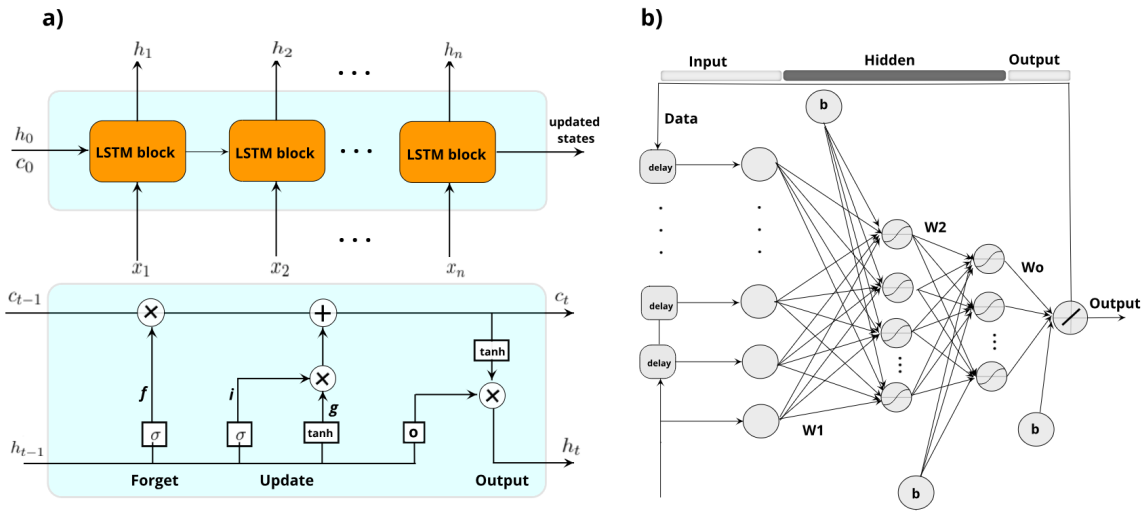

Figure 3. (a) Structure of a typical LSTM network; and (b) a typical NARX network with three layers. Linear and nonlinear (sigmoid function) activation functions are usually used for the output and hidden layers, respectively. b represents the bias, and W_0, W_1 and W_2 are weights at hidden/output layers. Neurons are connected by weights that their values become updated during the training phase of learning process.

$\sigma(x) = e^x/(1+e^x)$ and tanh is given by $(e^x - e^{-x})/(e^x + e^{-x})$. W_X are the weight matrices, b represents the bias. At each time step, the hidden state and the cell state, that remembers the previous values over arbitrary time period, are calculated as:

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t, \quad (4)$$

$$h_t = o_t \odot \sigma(c_t), \quad (5)$$

where $\odot$ represents the element-wise multiplication. In short, the first step in the LSTM identifies which data will be excluded from the cell via the sigmoid function by utilizing information of previous hidden layer h_{t-1} at time $t-1$ and the input x_t at present time t, i.e. the forget gate f_t. Deciding and storing information in the cell state use the sigmoid layer (deciding between omitting and updating) and the tanh layer (determining the level of importance of information by weighting them between $(-1, 1)$. The two values from deciding and updating steps are multiplied and then added to old memory c_{t-1} to obtain new cell state c_t. Final step creates the output values of h_t based on values of o_t and c_t using a sigmoid gate and a tanh layer, respectively, i.e. Eq. (4). The LSTM network as shown in Fig. 3-a has great ability to approximate the forecasting function g in Eq. (2). Furthermore, we standardize the data (i.e. zero mean and standard deviation of 1) before using them in the algorithm. This improves the effective learning and convergence rate of LSTM network, and shortens the training time of the network. In the case of multiple meteorological inputs, the normalization step makes the same time scales and structures for different input variables. In constructing LSTM model, one needs to carefully select training parameters such as learning rate, optimizer, and number of epochs. One epoch is equivalent with passing, only once, the data forward and backward through the network. In this study, we set different values of epochs to check the forecast error rate, and select "Adam" algorithm for optimization that employs a stochastic gradient descent procedure to update the weights of LSTM network [12]. Finally, the predictive ability of the LSTM can be substantially enhanced if the observed values are used for updates instead of predictions.

3.2. NARX

The learning process and convergence rate in the NARX are more effective/faster than other ANN techniques. Figure 3-b shows the structure of a standard NARX network including: a two-layer feed-forward network, delay lines to memorize the previous values of the input samples, a linear transfer function in the output layer, and a sigmoid transfer function in the hidden layer. The training data are passed through the delay lines. Here, the input meteorological data to the NARX are atmospheric pressure, air temperature, wind direction, i.e. exogenous input, as well as wind speed as endogenous input. The output is wind speed. This network gains from the memory ability (using previous values of predicted or true sample) that improves substantially its performance in the forecast scenarios.

For training of this network in order to update weights/biases, a back propagation methodology (based on computing gradients) is employed by the use of the Levenberg–Marquardt optimization. This optimization method minimizes a combination of the mean of the squared weights (biases) and the mean-squared errors for all neurons in the network to provide an estimate of g [13, 8]. As part of NARX architecture, the output at each time is fed back to the input of the network, Fig. 3-b. Moreover, feeding the target output during training, $\{u_t, \cdots, u_{t-t_u}\}$, can also improve the efficiency of the network optimization. It is noted that the NARX networks suffer from the vanishing gradient issue.

4. Results

Performance of forecasting models is evaluated through three metrics: The Nash–Sutcliffe Efficiency (NSE) [16], the Root Mean Square Error (RMSE), and the Mean Absolute Error (MAE) as:

$$\text{NSE} = 100 \times \left(1 - \frac{\sum_i (u_i - \hat{u}_i)^2}{\sum_i (u_i - \bar{u})^2}\right), \tag{6}$$

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(u_i - \hat{u}_i)^2}, \tag{7}$$

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|u_i - \hat{u}_i|, \tag{8}$$

where N denotes the total number of data, u_i, $\hat{u}_i$, and $\bar{u}$ present the observed, the predicted, and the mean wind speed, respectively at discrete time t_i. In this section, we aim to compare the performance and accuracy of NARX and LSTM (with update) neural network for 17-day-ahead forecast for Vigra station. The training sequence is selected for the first ∼90% of the time series, see Fig. 2-b. Here, the key difference between configuration of LSTM and NARX is associated with input time series as the NARX uses external data (i.e. air temperature, pressure, and wind direction) to improve forecast by accounting for external dependencies in addition to the wind speed.

Figure 4-a shows the predicted time series from the LSTM model along with absolute errors of forecast (Figs. 4-b and c) for the conditions where the network is not updated (i.e. LSTM-NU, red markers) and the network updated with observations (i.e. LSTM-WU, black line) against the wind speed observations (blue line). It is evident that with no-update, the prediction is quite acceptable for a few time steps (here about few hours). The RMSE error is 2.5 m/s and the absolute error has a peak with value of 6 m/s. After training of the LSTM network, we can replace the predicted values by the observed ones to update the LSTM network in the testing phase (when observations are available). This leads to a significant improvement in the predictions and notable reduction of RMSE error. The RMSE error becomes around 1 m/s and the maximum value of the absolute error reaches $\sim$ 4 m/s.

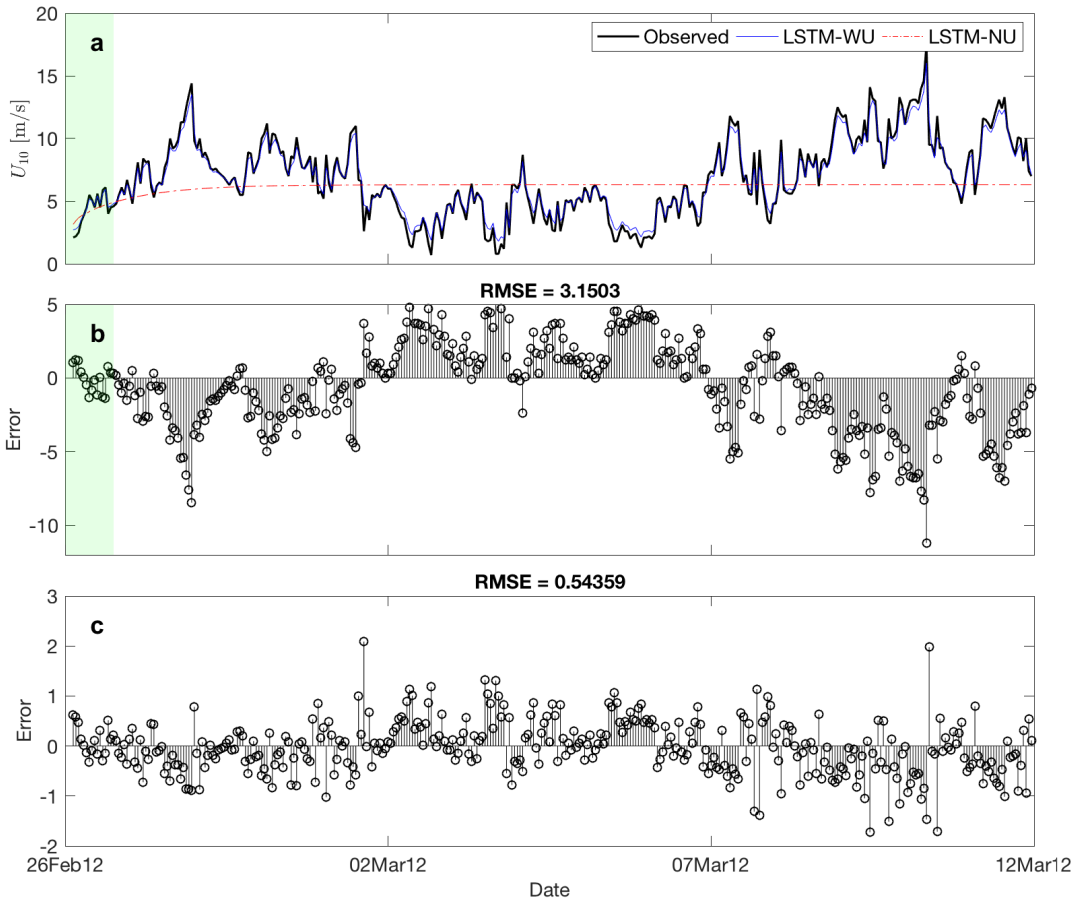

Figure 4. (a) Comparisons between observation (black) and prediction from LSTM-WU (blue) and LSTM-NU (red dotted line) in the test phase; (b) the absolute error of prediction from LSTM-NU; and (c) the absolute error of prediction from the LSTM-WU. The green shaded areas in (a) and (b) denote roughly the forecast horizon for the LSTM-NU.

Depending on the problem, the training error rates can be used as a criterion to determine near optimum values of hyper-parameters like number of epochs, number of neurons and hidden layers, and effects of the LSTM update with observations (i.e. LSTM-WU). In Fig. 5, we show comparisons between 4 LSTM simulations with the same values of hyper-parameters except different values of epochs. It is evident that increasing number of epochs will not necessarily lead to better quality of forecast. Specifically, the forecast for the LSTM-WU reaches better performance and accuracy when using epochs value of 60 (i.e. correlation coefficient of 0.89 and RMSE of 1.4 m/s), Figs. 5-a and b. The same can be concluded for the conditions when the neural network is not updated. In this case, we select only 18 hours of prediction for the purpose of comparison.

We use a NARX network with three hidden layers consisting of 40 neurons and set epoch to 60 in this study. This number of neurons has been selected based on trial and error to minimize the (mean-square) error (there is possibility to achieve satisfactory results by different network configuration and topology). Performance of network is calculated by the use of Eqs. (6-8), see

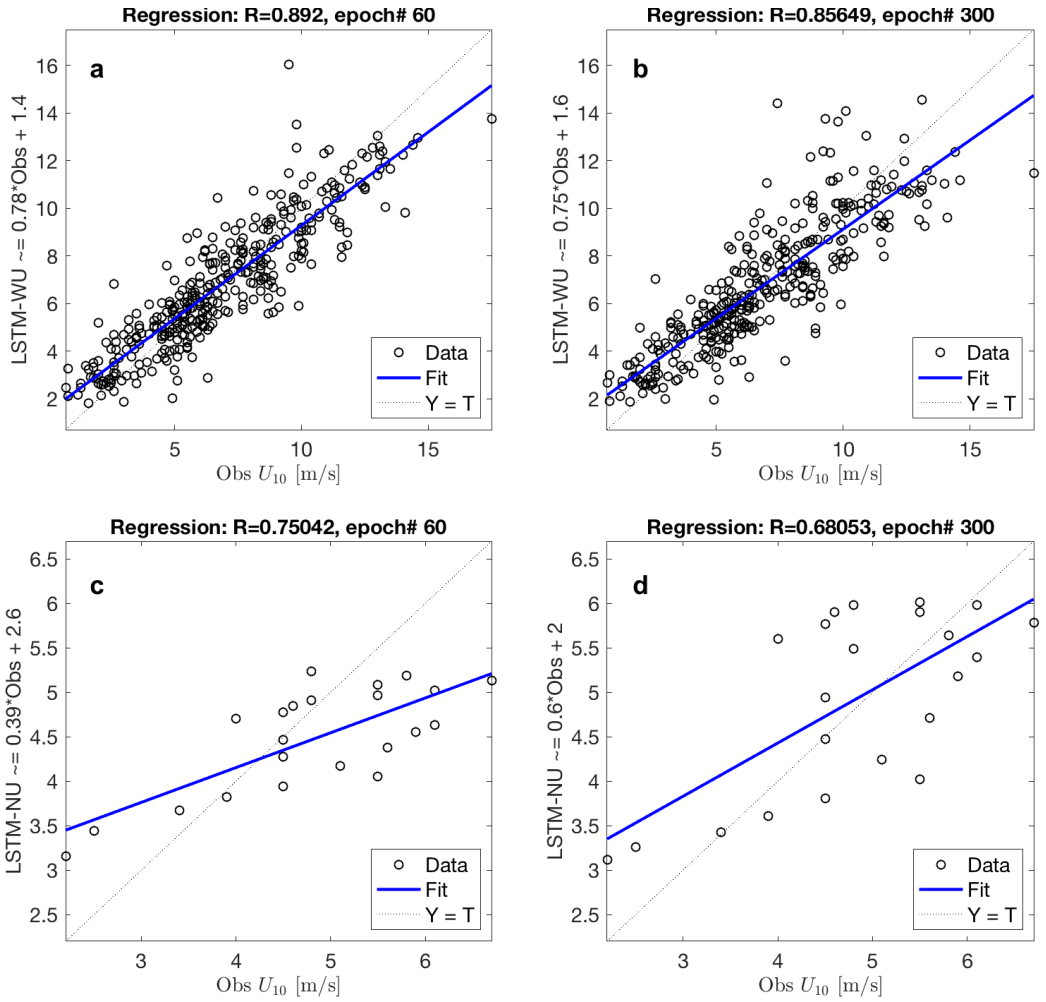

Figure 5. Scatterplots between observation and: (a) prediction from LSTM-WU with epochs number of 60; (b) prediction from LSTM-WU with epochs number of 300; (c) prediction from LSTM-NU with epochs number of 60; and (d) prediction from LSTM-NU with epochs number of 300. R denotes the correlation coefficient. Figures (c) and (d) are made for few time steps beyond the last recorded observations due to the short-horizon forecasting characteristics of LSTM-NU. For plots of LSTM-NU cases, we select only 18 hours of prediction.

Table 1. Learning rate and momentum hyper-parameters are set to 0.003 and 0.3, respectively. Finally, training algorithm uses Levenberg-Marquardt backpropagation training algorithm. We use hourly air temperature, atmospheric pressure, and wind direction with the same vector sizes as input and wind speed as output. Figure 6-a shows time series comparison between the measured and the predicted wind speed for $\sim$ 17 days. Two time series are highly in agreement with correlation value of 0.66. While the absolute error is low for the vast majority of points, there are few events that cause notable jump in the values of error, see the shaded green areas. In search to find out the reason for such behaviour, we noticed high correlation between this event and sharp variation in the wind direction data occurring at the same time. These events might

be caused by the wind gustiness or the lack of adequate quality data selection and treatment. To resolve the issue, we replaced the original wind direction time series with the low-passed filtered (smoothed) time series, Fig. 6-b black line. The effects of smoothing out such events has led to substantial improvement in the accuracy and performance of NARX as shown in Fig. 6-c.

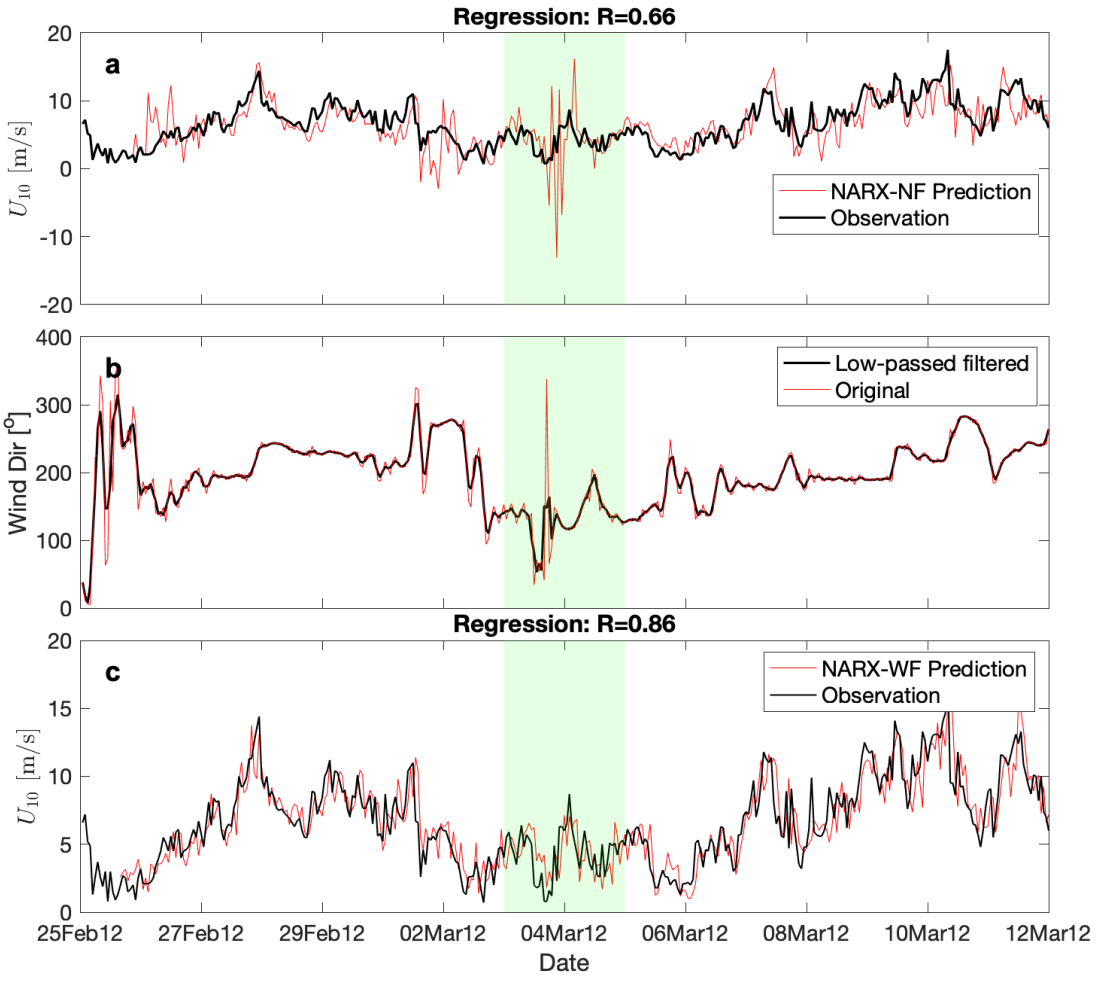

Figure 6. (a) Comparison between the observed (black) and the NARX-based prediction of the wind speed without smoothing the wind direction to damp out the steep variations (red); (b) comparison between the original (red) and the smoothed (black) wind directions; and (c) comparisons between the observed (black) and the NARX-based prediction of wind speed after applying low-passed filter to the wind direction input array (red). Here, R denotes correlation coefficient and epochs number sets to 60.

It is obvious from the scatterplots in Figs. 7-a and b that the NARX-WF (i.e. NARX with filter) produces better estimates of wind speed than the ones from the NARX-NF (i.e. NARX with no filter) so that the effect of low-passed filtering of wind direction leads to an increase of correlation between the observed and predicted wind speeds. The smoothing will further improve the pattern of correlation (i.e. scatters) between both time series. This example highlights importance of applying appropriate pre-processing procedures on input data to achieve

good feature expressions.

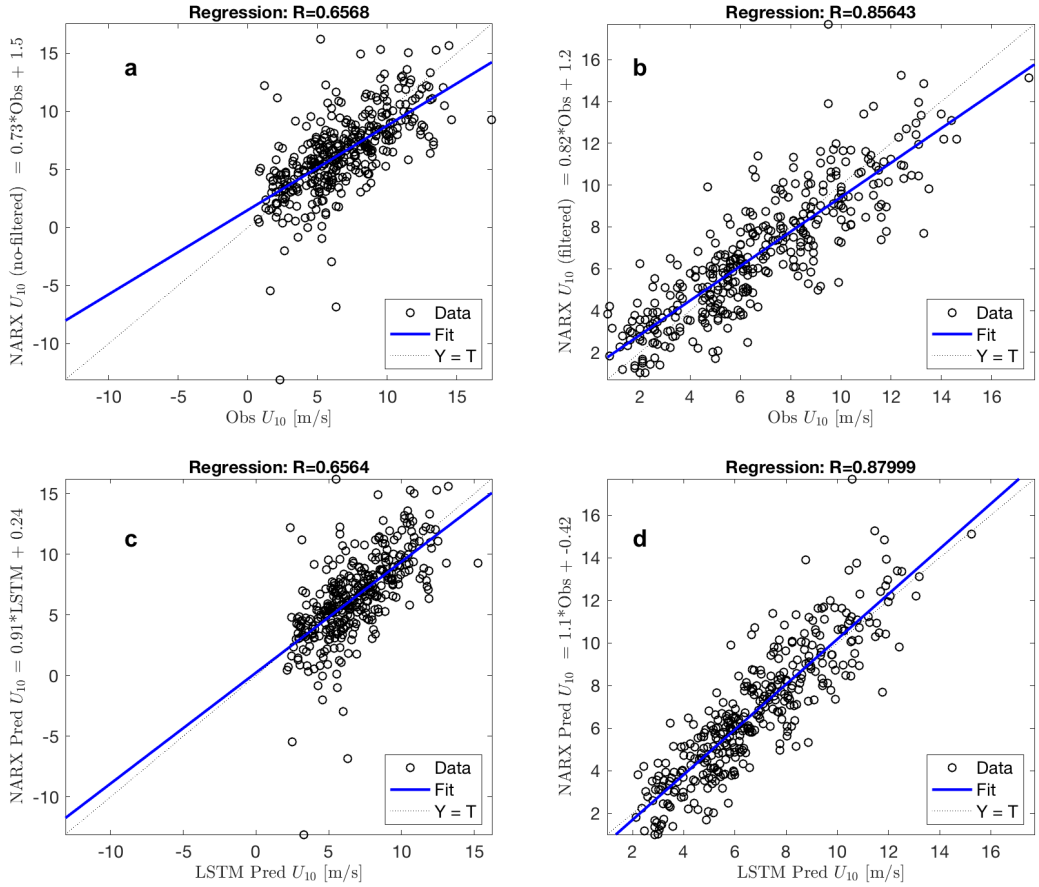

Figure 7. Regression analysis for following cases: (a) Between observation and the NARX prediction of U_{10} without smoothing (no low-passed filtering) of the wind direction, NARX-NF; (b) between observation and the NARX prediction with smoothing (with low-passed filtering), i.e. NARX-WF; (c) between the LSTM-WU prediction and the NARX-NF prediction; and (d) between the LSTM-WU prediction and the NARX-WF prediction. R denotes correlation coefficient, and vertical axes contain equations of regression lines. In this example, we set epochs number to 60 for both LSTM and NARX networks.

Table 1 quantifies further the comparison results between the predicted and the observed wind speed from the LSTM and the NARX networks as shown in Fig. 7. The accuracy of the LSTM model (NSE value) reaches well above 14% with an average error of 1.11 m/s (MAE value) and RMSE value of 0.68 m/s. This again suggests that the LSTM with update is the most accurate model in this study. Furthermore, smoothing the wind direction in the NARX model has a significant impact on the performance and accuracy of the NARX-based predictions.

5. Conclusions

In this study, we investigated the predictability of wind speed time series using two deep learning algorithms, i.e. Long Short Term Memory (LSTM) and Nonlinear Autoregressive with

Table 1. Quantitative comparisons of predictive models using performance and accuracy measures defined by Eqs. (6–8) in the testing phase.

		accuracy	
Method	RMSE [m/s]	NSE [%]	MAE [m/s]
LSTM-WU	0.676	14.22	1.113
NARX-NF	2.791	3.67	1.984
NARX-WF	1.663	7.00	1.267

EXogenous input (NARX). Both methods were trained and applied for the time series collected from a meteorological mast located in the close vicinity of Havsul-I offshore area in the west coast of Norway. While LSTM used only wind speed for the training and testing, the NARX used atmospheric pressure, air temperature, and wind direction as exogenous data in addition to the wind speed. We have shown that the LSTM model without update can predict successfully for at least a few time steps (i.e. short-term forecast with horizon of few hours), while updating the LSTM network with observation could significantly improve the long-term (i.e. more than a day) prediction. We showed that the NARX can successfully predict wind speed with high performance. It has been also observed that abrupt variability (e.g. from either strong wind gustiness or lack of sufficient accuracy of temporal features in data) can decline the accuracy and performance of the NARX model. The issue was resolved by using the wind direction data at low frequencies during the learning process (to avoid very extreme variations between local minima and maxima). A shortcoming associated with the forecast using NARX in this study is the lack of using the solar irradiation data as input for the learning process. This is because adding good quality solar irradiation data provides further information on regionality and diurnal variability that could potentially improve effective prediction of wind speed.

To validate all model results, we applied some criteria such as NSE value, RMSE value, as well as mean absolute error value. Both updated LSTM and NARX showed very good performance in forecasting wind speed using test data. However, the updated LSTM outperformed the NARX model. It is worth to mention, the accuracy and performance of single-input LSTM model without update in forecasting of more complex wind speed time series will increase if we employ multiple meteorological variables as input data or integrate LSTM with other data-driven or statistical models.

5.1. Acknowledgments

This study has awarded funding from the academic agreement between UiB and Equinor.

References

[1] U.S. Department of Energy. *Staff Report to the Secretary on Electricity Markets and Reliability*, 2017. Available online: https://www.energy.gov/downloads/download-staff-report-secretary-electricity-markets-and- reliability.

[2] Tsikalakis A G, et al., 2009 Impact of wind power forecasting error bias on the economic operation of autonomous power systems. *Wind Energy*, **12(4)**, 315-331.

[3] Carlini E M, et al., 2016 Physical and statistical downscaling for wind power forecasting. *In: Proc. of 2016 International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM)*,

[4] Barbounis T G, Theocharis J B, Alexiadis M C, Dokopoulo P S, 2006 Long-term wind speed and power forecasting using local recurrent neural network models. *IEEE Trans. Energy Convers.*, **21(1)**, 273–284.

[5] Cao Q, Ewing B T, Thompson M A, 2012 Forecasting wind speed with recurrent neural networks. *Eur. J. Oper. Res.*, **221(1)**, 148–154.

[6] Kariniotakis, G N, Stavrakakis G S, and Nogaret E F, 1996 Wind power forecasting using advanced neural networks models. *IEEE Transactions on Energy Conversion*, **11(4)**, 762-767.
[7] Cadenas E and Rivera W, 2010 Wind speed forecasting in three different regions of Mexico, using a hybrid ARIMA-ANN model. *Journal of Renewable Energy*, **35(12)**, 2732–2738.
[8] Cadenas E, Rivera W, Campos-Amezcua R, and Heard C, 2016 Wind speed prediction using a univariate ARIMA model and a multivariate NARX model. *Energies*, **9(2)**, 109.
[9] Jabbari A and Bae D H, 2018 Application of Artificial Neural Networks for Accuracy Enhancements of Real-Time Flood Forecasting in the Imjin Basin. *Water*, **10**, 1626, doi:10.3390/w10111626.
[10] Jiang G-Q, Xu, J, Wei J, 2018 A deep learning algorithm of neural net- work for the parameterization of typhoon-ocean feedback in typhoon forecast models. *Geophysical Research Letters*, **45**, 3706–3716.
[11] Hallas M, Dorffner G, 1998 A comparative study on feedforward and recurrent neural networks in time series prediction using gradient descent learning.
[12] Hochreiter S, Schmidhuber J, 1997 Long short-term memory. *Neural Comput.*, **9(8)**, 1735–1780.
[13] Lin T, Horne B G, Tino P, and Giles C L, 1996 Learning long-term dependencies in NARX recurrent neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, **7(6)**, 1329–1338.
[14] Bakhoday-Paskyabi M and Fer I, 2014 Turbulence structure in the upper ocean: a comparative study of observations and modeling. *Ocean Dyn.*, **64**, 611-631.
[15] Pascanu R, Mikolov T, Bengio Y, 2013 On the difficulty of training recurrent neural networks. *In International Conference on Machine Learning*, 1310–1318.
[16] Nash J E and Sutcliffe J V, 1970 River flow forecasting through conceptual models part I-A discussion of principles. *Journal of Hydrology.*, **10(3)**, 282–290, doi:10.1016/0022-1694(70)90255-6