Bias correction of wind power forecasts with SCADA data and continuous learning

Jonas, S.; Winter, K.; Brodbeck, B.; Meyer, A.

DOI
10.1088/1742-6596/2767/9/092061

Publication date
2024

Document Version
Final published version

Published in
Journal of Physics: Conference Series

Journal of Physics: Conference Series

PAPER • **OPEN ACCESS**

Bias correction of wind power forecasts with SCADA data and continuous learning

To cite this article: S Jonas *et al* 2024 *J. Phys.: Conf. Ser.* **2767** 092061

Journal of Physics: Conference Series **2767** (2024) 092061 doi:10.1088/1742-6596/2767/9/092061

Bias correction of wind power forecasts with SCADA data and continuous learning

S Jonas[1,2], K Winter[3], B Brodbeck[3] and A Meyer[1,4]

[1] School of Engineering and Computer Science, Bern University of Applied Sciences, Quellgasse 12, 2501 Biel, Switzerland

[2] Faculty of Informatics, Università della Svizzera italiana, Via la Santa 1, 6962 Lugano-Viganello, Switzerland

[3] WinJi AG, Badenerstrasse 808, 8048 Zurich, Switzerland

[4] Department of Geoscience and Remote Sensing, Delft University of Technology, Stevinweg 1, 2628 Delft, Netherlands

E-mail: stefan.jonas@bfh.ch

Abstract. Wind energy plays a critical role in the transition towards renewable energy sources. However, the uncertainty and variability of wind can impede its full potential and the necessary growth of wind power capacity. To mitigate these challenges, wind power forecasting methods are employed for applications in power management, electricity trading, or maintenance scheduling. In this work, we present, evaluate, and compare four machine learning-based wind power forecasting models. Our models correct and improve 48-hour forecasts extracted from a numerical weather prediction (NWP) model. The models are evaluated on datasets from a wind park comprising 65 wind turbines. The best improvement in forecasting error and mean bias was achieved by a convolutional neural network, reducing the average NRMSE down to 22%, coupled with a significant reduction in mean bias, compared to a NRMSE of 35% from the strongly biased baseline model using uncorrected NWP forecasts. Our findings further indicate that changes to neural network architectures play a minor role in affecting the forecasting performance, and that future research should rather investigate changes in the model pipeline. Moreover, we introduce a continuous learning strategy, which is shown to achieve the highest forecasting performance improvements when new data is made available.

1. Introduction

Recent growth and progress of renewable energy sources, especially solar and wind, have caused rapid transformative changes in power systems across the world [1]. Wind energy in particular, the leading non-hydro renewable energy source, has accounted for a record growth of 17% in 2021 by 273 TWh, the highest growth amongst all renewables [2]. However, as wind energy develops into a more critical part of energy grids worldwide, so does the necessity for improved wind power forecasting [3]. Generally, the uses of wind speed and power forecasting can be divided into subgroups by the considered forecasting time horizon [4]: Very short-term (up to 30 minutes) forecasts can be employed for turbine control and load tracking, a short-term (up to 6 hours) horizon with applications for preload sharing, medium-term (6 to 24 hours) for power system management and electricity trading, and long-term (1-7 days) forecasting used for purposes such as maintenance scheduling, the most relevant to our presented work, in which we will present wind power forecasting models with a forecast horizon of 48 hours.

Journal of Physics: Conference Series **2767** (2024) 092061
doi:10.1088/1742-6596/2767/9/092061

1.1. Previous work in wind power bias correction

The wind power forecasting task can be split into methods based on physical models and statistical models [4]: Physical models, such as numerical weather prediction (NWP) models, provide forecasts for wind speeds and other meteorological variables by computationally solving physical equations describing atmospheric processes. These models are computationally expensive and typically provide a limited resolution. Forecast data derived from NWP models can contain inherent uncertainties and biases, coming from the model formulation, simplification of physics, initial measurements, or surface characteristics [5]. Statistical methods aim to model the relationship between the target variable (e.g., wind turbine power output) and the available observed data using statistical (learning) techniques. For an overview of advances in statistical methods, we refer the reader to the reviews of [4, 6]. Another subcategory of statistical approaches, into which our work falls into, are statistical methods which incorporate NWP model forecasts as predictor variables. The purpose of these hybrid models is to *correct* any biases inherent in the NWP model and to enhance the forecasting capabilities of the provided forecasts. Early works such as [7] have shown that incorporating NWP data as predictor variables can greatly improve hour-ahead forecasts compared to pure statistical models. The results discussed in [8] suggest that corrected NWP data from a model with a moderate resolution can provide more reliable forecasts than obtaining computationally expensive high resolution NWP data. Examples of the present bias found in NWP data are discussed in [9, 10, 6]. The NWP errors are shown to be systematically linked to the season and the wind speed and direction. Similarly, [11, 12] extract specific NWP error patterns and incorporate these groups of errors into their correction models. A prominent correction approach present in earlier works is Kalman filtering [8, 13, 14]. Gaussian process regression can improve upon this approach by incorporating multiple input variables into the model, such as the NWP wind speed and direction, humidity, temperature, or atmospheric stability variables, as in [15, 16]. Deep neural networks as forecasting models have recently become prevalent. A standard artificial neural network is presented in [11, 17, 18], where in the latter a novel look-back parameter is introduced to determine how many hours of historically observed wind speed to include in the input variables. LSTM or RNN-based architectures are very commonly used as basis [19, 20, 21, 22, 23]. In [24], a novel architecture is introduced inspired by denoising autoencoders, further confirming the benefit of incorporating atmospheric variables as input features. A more advanced deep learning architecture was presented in [25]. An encoder-decoder based temporal-attention network is proposed, which learns to selectively choose data from the multi-source NWP dataset and the extent of historical data to include.

1.2. Contribution to research

Despite these numerous advancements, the insights remain limited: As the comparability across works is naturally restricted due to differences in turbines, locations, datasets, and forecast horizons, the best approach remains unclear. The presented results tend to not be adequately compared to other benchmarks. We provide such a comparison of four commonly used and yet significantly different machine learning-based models. Moreover, we introduce continuous learning to the wind power forecasting task. With continuous learning we refer to a strategy of periodic updating of the model whenever new data has been made available. To the best of the author's knowledge, we are the first to investigate the best strategies to update a wind power forecasting model to the newest data.

2. Datasets and case study

2.1. Acquired data

Our dataset contains measurement data of 65 commercial 2.1 MW wind turbines (WTs) from a wind farm in which all turbines share the same manufacturer and technical specifications.

Journal of Physics: Conference Series **2767** (2024) 092061 doi:10.1088/1742-6596/2767/9/092061

The Supervisory Control and Data Acquisition (SCADA) data provides for each WT 10-minute averages of the measured turbine power output, wind speed, nacelle and wind direction, and the environment temperature for a continuous timeframe of 2 years. We obtained Numerical Weather Prediction (NWP) data based on the Weather Research and Forecasting model (WRF [26]) for the identical location as the wind farm and for the same timeframe. This data comprises computed forecasts of the wind speed, wind gust, environment temperature, wind direction, radiance, and precipitation. The forecasts have a lead time for up to 72-hours, at an interval of 15 minutes, and are re-calculated every 6 hours.

2.2. Data processing

Feature selection and pre-processing. The forecasts for wind speed, wind gust, wind direction, and temperature from the NWP forecast dataset were retained. From the SCADA dataset, we extracted only the measured turbine power output as target variable. We adjusted all NWP wind speed values to the hub height of the turbines by applying a log-adjustment: $WS_{NWP*} = WS_{NWP} * \frac{log(h_{HUB})}{log(h_{NWP})}$, where h_{NWP} and h_{HUB} are the forecast height for the wind speed by the NWP model and the turbine hub height respectively. For each timestep t, the following 10 variables were included as predictors: NWP wind speed, NWP wind gust, NWP temperature, NWP wind direction (sin, cos), hour of t as cyclical variables (sin, cos), t as day of year as cyclical variables (sin, cos), and the forecast lead time. We normalized each variable separately, such that each feature value is in the range of $[0, 1]$.

Data matching. The NWP and SCADA data were matched so as to create K samples of 48-hour forecasts for each wind turbine separately, where each forecast sample, starting at the unique time t_0, is composed of the selected 10 NWP-based input features and the observed turbine power outputs for times $\{t_0, t_{0+1h} \ldots, t_{0+48h}\}$. A forecast sample k consists of the 10 input features based on the NWP forecast data $\mathcal{X}$ for each timestep, and the corresponding true power outputs $\mathcal{Y}$ (target variable): $\mathcal{X}_k = \{NWP_{t_0}\}, \ldots, \{NWP_{t_{0+48h}}\}$, $\mathcal{Y}_k = \{P_{t_0}, \ldots, P_{t_{0+48h}}\}$, where NWP_t are the 10 processed NWP-based input features for that specific timestep, and P_t is the observed turbine power output for time t.

Dataset split. The forecast samples were then split into a non-overlapping training-, validation- and test set. The split was performed monthly, for which randomly chosen 20% of consecutive days were assigned to the validation set, 20% of consecutive days to the test set, and the remaining days to the training set. Each turbine's dataset was split identically.

3. Models

3.1. Wind power forecast models

We introduce four machine learning-based correction models trained to correct the biases and to improve the forecast capabilities compared to an uncorrected baseline model. The models are trained for all 65 WTs in the dataset separately. In general, each of the four presented correction models takes as input the pre-processed NWP data and outputs a wind power forecast. The gradient boosting regression model (GB) and the artificial neural network (NN) output a single wind power forecast for time t, based on the NWP features for only the specified single timepoint, while the convolutional neural network (CNN) and the long short-term memory (LSTM) models predict all 48 consecutive hours of wind power data at once, based on the entire 48 hours of NWP features as input sequence in the forecasting sample. Formally, the two single-timepoint models GB and NN are of the form: $f\left(X_{k,t}\right) = \widehat{Y_{k,t}}; \qquad k = 1, \ldots, K; \quad t = t_0, \ldots, t_0 + 48h,$ whereas the CNN and LSTM models output a sequence of power forecasts per sample: $f\left(X_k\right) = \widehat{Y_k}; \qquad k = 1, \ldots, K$

Baseline. The baseline model represents the uncorrected forecasts based on the NWP data, in our study it is a power curve provided by the WT manufacturer. The model takes as input the NWP wind speed forecast for a timepoint t and outputs an expected power output. As the input is not modified, it transfers all the biases inherent in the NWP model to the power output, i.e., it is an uncorrected forecast. This model is used to assess the extent of the bias in the power prediction task and to evaluate the improvements achieved by the other presented approaches. We compare against this baseline to assess the improvement potential over a WRF-derived wind and power forecast using machine learning. Future work may involve including a wake model into the NWP baseline model (e.g., [27]).

Gradient boosting. Gradient boosting [28] is a popular machine learning-based technique to build prediction models. In this work, we propose a decision tree-based gradient boosting regression model. This model outputs a wind power prediction for a timepoint t, given the ten corresponding input features extracted from NWP data. The model aims to correct the NWP forecasts and to output an unbiased wind power prediction. Learning is performed by minimizing the mean squared loss between the model output and the observed WT power. We trained a separate model for each WT, i.e., 65 individual gradient boosting regression models were developed. We used 100 boosting stages with a max depth of 5, and a learning rate of 0.05.

Neural network. The fully-connected neural network (NN) aims to learn to output a corrected wind power forecast given the ten corresponding input features of time t. This network consists of three fully-connected layers (64, 64, 1 units, respectively). During training, the weights of the neural network are optimized by minimizing the mean squared error loss between the true power outputs and the network outputs. In doing so, the network learns the relationship between the NWP input features and the expected turbine power output.

Long Short-Term Memory Network. This model is based on a long short-term memory network architecture (LSTM) [29]. Specifically, we propose a model using a bidirectional LSTM layer [30], followed by three fully-connected layers (16, 32, and 49 units, respectively). LSTM architectures are ideally suited for sequences of data. As such, and unlike the previously presented models, the LSTM input is a sequence of features from the entire 48-hour forecast sample and the model outputs a power prediction for all timepoints in that sequence. The LSTM layer additionally attempts to learn information present *within* the sequence, i.e., between features of different timepoints. A bidirectional layer can extract information in the forward sequence direction (from oldest forecasts to newest) as well as in the backward direction.

Convolutional Neural Network. Convolutional Neural Networks (CNNs) are prevalent architectures particularly successful in computer vision tasks [31]. In this work, we propose a network based on 1-D convolutional layers, where the input is a 1-D timeseries consisting of the sample timesteps t_0 to t_{0+48h} (width of the grid) and the corresponding ten input features of each timepoint (channels). The network consists of two 1-D convolutional layers, followed by two fully-connected layers (96 and 49 units, respectively). Similar to the LSTM network, the CNN aims to learn temporal relationships between input features at different timepoints.

3.2. Model selection
The model hyperparameters were determined through a random search algorithm using one randomly selected WT. We evaluated 200 different model configurations for each of the four model architectures. The configurations with the lowest root mean squared error calculated on the WT-specific validation set were chosen as optimal hyperparameters.

3.3. Training procedure and implementation

All models were trained separately on the training set of each wind turbine. That is, we trained 4 WT-specific correction models for all 65 WTs (260 models). During training, the mean squared error between the model output (power prediction) and the true observed turbine power output was minimized. Early stopping was implemented for the neural networks to stop training after the validation loss had not improved within 15 epochs. This work was implemented in Python v3.10 and Keras v2.11 [32]. All experiments were run on an Intel Xeon CPU @ 2.3 GHz and a NVIDIA Tesla T4 GPU. The required times to train one single model were as follows (CPU/GPU): GB: $274s/-$, NN: $189s/187s$, CNN: $120s/44s$, LSTM: $968s/66s$.

4. Results and discussion

4.1. Bias analysis

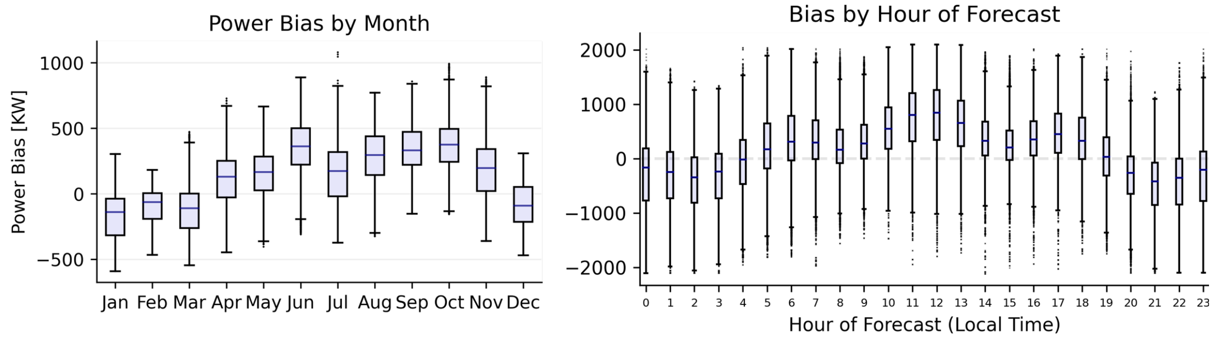

Figure 1. Bias (predicted - observed power) of the uncorrected baseline model in relation to the month (left) and the hour of forecast (right).

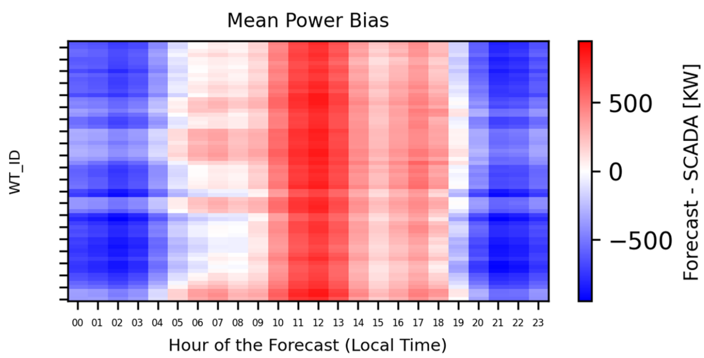

Figure 2. Mean Bias (predicted - observed power) of the uncorrected baseline model across all 65 WTs in relation to the hour of forecast.

Figure 1 shows two examples of significant power bias resulting from the power forecasts of the baseline model for one WT. The bias is calculated as the difference between the baseline prediction, i.e., from plugging in the uncorrected NWP wind speed forecasts into the provided power curve, and the true power output from the SCADA data. The figure on the left shows a varying power bias depending on the season of the year, where the power is systematically underestimated (negative mean power bias) by the NWP model in the months December, January, and February, and strongly overestimated in the months June, July, and August. On the right-hand side, an association between the bias and the local hour of the time is observed, where the power is strongly overestimated during daytime hours (6-18) and conversely strongly underestimated during nighttime hours. Multiple factors can contribute to these biases, such as the parameterisation of processes in the atmospheric boundary layer. The model may not accurately represent diurnal and seasonal aspects of boundary layer processes and boundary layer height, convection, turbulence and mixing at the wind farm site. This may be mitigated by adjusting boundary layer parameterisations. The model's representation of land surface characteristics related to topography, surface roughness and vegetation may also contribute to biases in simulated wind patterns. These results support the inclusion multiple input features, rather than just correcting the wind speed. Further, we observed different exhibitions of the bias across wind turbines as shown in Figure 2, suggesting the need for individually fitted models rather than one global model. These individual differences can for instance be due to geospatial differences, individual turbine characteristics (e.g., maintenance history, age), or wake effects.

4.2. Evaluation metrics

We trained all proposed models from Section 3 on the training sets and evaluated them on the test sets of each WT. The evaluation data based on K 48h-forecast samples contain the model power predictions $\widehat{Y_{t_0,k}}, \dots, \widehat{Y_{t_0+48,k}}$, and the corresponding true power outputs $Y_{t_{0_k}}, \dots, Y_{t_0+48,k}$ for each forecast sample k. We assess the model performance with the mean bias (MB), the mean absolute error (MAE), and the root mean squared error (RMSE) as follows:

$$Mean\ Bias = \frac{1}{K}\sum_K MB_k = \frac{1}{K}\sum_K \frac{\sum_T \left(\widehat{Y_{t,k}} - Y_{t,k}\right)}{T} \tag{1}$$

$$MAE = \frac{1}{K}\sum_K MAE_k = \frac{1}{K}\sum_K \frac{\sum_T \left(|\widehat{Y}_{t,k} - Y_{t,k}|\right)}{T} \tag{2}$$

$$RMSE = \frac{1}{K}\sum_K RMSE_k = \frac{1}{K}\sum_K \sqrt{\frac{\sum_T \left(\widehat{Y_{t,k}} - Y_{t,k}\right)^2}{T}} \tag{3}$$

$$NRMSE = \frac{1}{K}\sum_K RMSE_k \times \frac{100}{C} = \frac{1}{K}\sum_K \sqrt{\frac{\sum_T \left(\widehat{Y_{t,k}} - Y_{t,k}\right)^2}{T}} \times \frac{100}{C} \tag{4}$$

where $\widehat{Y_{t,k}}, Y_{t,k}$ represent the model power prediction and the true turbine power output of forecast sample k at timepoint t, respectively, and C represents the installed capacity of the wind turbines.

4.3. Model evaluations

Our four model architectures were evaluated and compared on the test set of each turbine separately. In Table 1 the averages of the mean bias, MAE, RMSE, and NRMSE obtained on the test sets of all 65 WTs are listed. The baseline represents the uncorrected forecasting scenario, i.e., the power predictions obtained from plugging in the uncorrected NWP wind speed forecast into the power curve. With an average mean bias of -67.2 KW (+- 86), the baseline model shows a very significant bias, as already indicated in the bias analysis of Figure 1. Additionally, the mean RMSE obtained from the baseline model is at a very high value of 725 KW (34.5% NRMSE), implying unreliable forecasts. Our four machine learning-based correction models all show a significant reduction in bias and errors on the test sets: The gradient boosting regression model improves the average RMSE by more than 200 KW and shows a reduced but still considerable mean bias of 51 KW (+- 30). This model exhibits the least correction capabilities in all metrics amongst the other three correction models. The neural network architectures (NN, CNN, LSTM) improved the power forecasting significantly better: The lowest bias was achieved by the convolutional network, at only 12 KW (+- 27). Additionally, it significantly improved the forecasts down to an average RMSE of 463 KW (NRMSE 22%). In Figure 3, an example of the bias correction capabilities by the CNN is shown. Compared to the baseline, which shows a strong systematic deviation based on day- or night-time, the CNN forecasts are corrected, with mean biases close to 0 (unbiased), independent of the forecast hour.

Table 1. Mean test set forecast performance across the test sets of all 65 WTs.

Model	Avg. Mean Bias KW	Avg MAE KW	Avg. RMSE KW	Avg. NRMSE %
Baseline	-67.2 [+- 85.8]	585.5 [+-23.4]	725.1 [+-27.3]	34.5 [+- 1.3]
Gradient Boosting	50.5 [+- 30.4]	435.8 [+-15.1]	519.1 [+-27.3]	24.7 [+- 0.8]
Neural Network	39.8 [+- 22.2]	393.2 [+-13.4]	476.9 [+-16.9]	22.7 [+- 0.8]
CNN	**11.7 [+- 26.5]**	**375.3 [+-14.9]**	462.9 [+-16.8]	**22.0 [+- 0.8]**
LSTM	44.3 [+- 32.9]	385.6 [+-19.0]	**462.3 [+-19.8]**	**22.0 [+- 0.8]**

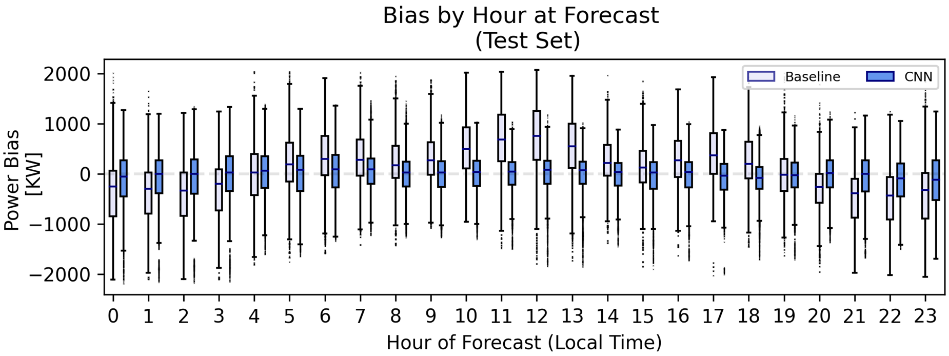

Figure 3. Bias (prediction - observed) of the uncorrected baseline model (grey) in comparison to the bias of the corrected CNN model (blue) by hour of forecast.

4.4. Model comparisons

There is a limitation in comparing the obtained absolute error values across WTs, as the forecast quality of the NWP wind speed forecasts can significantly differ across each WT. This discrepancy can be attributed for instance to individual turbine differences (e.g., maintenance history, age), terrain characteristics, or wake effects. Instead, we compare the forecasting error reduction in *relative* terms of the WT-specific baseline (uncorrected model), as shown in Table 2. These results first confirm the previous results, showing the least improvement by the gradient boosting regression model at an average 71.6% (28.4% error reduction) of the baseline RMSE, and the neural networks performing better and closely to each other, achieving 63.8 – 65.8% RMSE (an improvement of 34.2 – 36.2%) of the baseline. The independently trained models across all 65 WTs perform similarly and consistently, as implied by the minor standard deviation of less than 3% for all models.

We note that these four model architectures are considerably different to each other. The gradient boosting model has shown to result in considerably higher forecasting errors compared to all neural network models, and with an average rank of 4.0 out of 4, our results further advocate for the already prevalent use of neural network-based methods for the power forecasting task. However, we notice that the three different neural networks produce highly similar results. The fully-connected neural network architecture, trained on single timepoints and consisting of only 5,185 trainable parameters, produces only slightly higher RMSE errors (2% higher relative to the baseline) compared to the other neural networks. The CNN and LSTM models, on the other hand, are trained on complete 48h forecast sequences and contain 81,593 and 85,937 trainable parameters, respectively. Consequently, the benefit of learning the relationship of features across timepoints was only minimal in our case study. Regardless of the completely different architectures, these two models perform almost identically in terms of the forecasting ability. Our results hint at a limit of the possible bias correction and forecasting improvements with the presented pipeline of correcting NWP-based forecasts. Instead of investigating novel model architectures, future research directions should thus rather include conceptual changes, as has been already successfully shown by including previous measurements in the input features (e.g., [15, 18, 25]) or incorporating information from the power curve (e.g., [15, 18]).

Table 2. Mean forecasting performance in relation to the WT-specific baselines.

Model	Avg. MAE in % of Baseline	Avg. RMSE in % of Baseline	Avg. Rank by lowest RMSE
Gradient Boosting	74.5 [+- 2.8]	71.6 [+-2.6]	4.0 [+-0.0]
Neural Network	67.2 [+- 2.5]	65.8 [+-2.3]	2.8 [+-0.4]
CNN	**64.2 [+- 2.7]**	63.9 [+-2.3]	1.6 [+-0.6]
LSTM	65.9 [+- 2.6]	**63.8 [+-2.3]**	**1.5 [+-0.7]**

4.5. Continuous learning

A dataset consisting of collected data during 2 years may lack examples for certain meteorological events (e.g., heatwaves) which did not occur in the available timeframe. Trained models on these data could therefore generalize poorly when predicting based on data originating from previously unseen circumstances. There is a clear benefit to update the models to any newly made available data in order to introduce more examples to the model. We introduce the concept of continuous learning (also continual- or lifelong learning) [33], as a strategy to continuously update correction models. Specifically, we propose a procedure similar to the concepts of freezing and fine-tuning weights in transfer learning tasks (e.g., [34]), in which the original weights of the neural networks are used as the initial state, following a slight adjustment ("fine-tuning") of a subset of layer weights (non-"frozen" layers). With continuous learning, when given new training data, a subset of layer weights is trained on exclusively new data with a smaller learning rate, in order to update the models to new examples, while retaining the learned relationships from the original dataset. As the presented continuous learning strategy relies on neural network weights, it is not applicable to the gradient boosting model. We assess the effectiveness of our continuous learning strategy. New data comprising 6 months of forecasts and observations for all wind turbines were collected. The datasets were processed according to the same procedure as the original datasets. We evaluated the forecasting performance on the new test set of one randomly selected WT for the following three strategies:

Strategy 1 – original models: The models (NN, LSTM, CNN) remain unchanged and are evaluated on the test set of the new timeframe.

Strategy 2 – new models: We train the three neural networks from scratch exclusively on the new dataset, that is, only on the training set of the new 6 months of our updated dataset. As

the training and test set originate from the same timeframe, both sets should have a significant overlap in underlying weather and turbine conditions.
Strategy 3 – continuous learning: This strategy represents our continuous learning approach. Using the original neural network weights as initial state, we fine-tune selected layer weights by training with the new dataset using a smaller learning rate.

The results are shown in Table 3. We show the RMSE obtained on the new test set of the selected WT. For the three neural network architectures, we observe the same pattern across each model: Strategy 1 results in the highest RMSE, while in comparison strategy 2 shows a slight reduction in measured error. Our continuous learning approach reaches the best performance in all three cases. As the first strategy uses the original models, these results indicate that there may be some patterns present in the new data which did not occur in the original timeframe, leading to a higher forecast error compared to the other strategies. The improvements in performance achieved by the second strategy show the importance of having the same underlying conditions present in the training data: Despite being trained on significantly less data, the models trained with this strategy show a superior performance to the original models. With our continuous learning strategy, we adjust the pre-trained model weights of the neural networks to the new data, resulting in the models adjusting the previously learned feature relationships to new training instances. Our findings support the use of a continuous learning strategy, with which the models could be iteratively updated whenever new data becomes available, in order to improve the forecast and correction abilities of the models. However, future studies are required to confirm these results and to investigate potential substantial downsides of continuous learning such as catastrophic forgetting [33].

	RMSE [KW]		
Strategy / Model	NN	LSTM	CNN
Strategy 1	454	435	436
Strategy 2	437	431	425
Strategy 3	**435**	**408**	**416**

Table 3. Comparison of the strategy performances on the updated test set of a randomly selected WT.

5. Conclusion

In this work we investigated the task of wind power forecasting based on data extracted from a numerical weather prediction model. The baseline model using uncorrected NWP data was observed to output a strongly biased wind power forecast. Our analysis revealed a strong systematic over- and underestimation based on the seasonality and the time of day, suggesting the necessity of a correction model. We proposed four machine learning-based models to correct biases, namely a gradient boosting regression model, a fully-connected neural network, a long short-term memory network, and a convolutional neural network. All four correction models managed to significantly improve the baseline performance. The best forecasting performance was achieved by the convolutional neural network, although the very close performance proximity between the networks, despite being made up of completely different configurations, indicates a limit of the possibility of further improving wind power forecasts with the presented forecasting pipeline. Our results indicate that future research directions should investigate changes to the model pipeline, such as including different features (e.g., historical SCADA data) or more data sources. Finally, we introduced continuous learning, a strategy to continuously update models when new data becomes available. Our continuous learning strategy proved to achieve the best performance when updated and tested on new data. Further studies with larger datasets are required to assess the benefits and limits of continuous learning.

Journal of Physics: Conference Series **2767** (2024) 092061 doi:10.1088/1742-6596/2767/9/092061

Acknowledgments
We gratefully acknowledge the Swiss Innovation Agency Innosuisse for funding this project.

References
[1] IEA 2022 Renewables Tech. rep. IEA Paris URL `https://www.iea.org/reports/renewables`
[2] IEA 2022 Wind Electricity Tech. rep. IEA Paris URL `https://www.iea.org/reports/wind-electricity`
[3] Haupt S E and Mahoney W P 2015 *IEEE Spectrum* **52** 47–52 number: 11 Publisher: IEEE
[4] Jung J and Broadwater R P 2014 *Renewable and Sustainable Energy Reviews* **31** 762–777 ISSN 13640321
[5] Al-Yahyai S, Charabi Y and Gastli A 2010 *Renewable and Sustainable Energy Reviews* **14** 3192–3198 ISSN 13640321 number: 9
[6] Wang H, Yan J, Liu Y, Han S, Li L and Zhao J 2017 *J. Phys.: Conf. Ser.* **926** 012007 ISSN 1742-6588, 1742-6596
[7] Larson K A and Westrick K 2006 *Wind Energ.* **9** 55–62 ISSN 1095-4244, 1099-1824 number: 1-2
[8] Louka P, Galanis G, Siebert N, Kariniotakis G, Katsafados P, Pytharoulis I and Kallos G 2008 *Journal of Wind Engineering and Industrial Aerodynamics* **96** 2348–2362 ISSN 01676105 number: 12
[9] Yan J, Liu Y, Han S and Qiu M 2013 *Renewable and Sustainable Energy Reviews* **27** 613–621 ISSN 13640321
[10] Pearre N S and Swan L G 2018 *Sustainable Energy Technologies and Assessments* **27** 180–191 ISSN 22131388
[11] Xu Q, He D, Zhang N, Kang C, Xia Q, Bai J and Huang J 2015 *IEEE Trans. Sustain. Energy* **6** 1283–1291 ISSN 1949-3029, 1949-3037 number: 4
[12] Qu G, Mei J and He D 2013 *2013 11th IEEE International Conference on Industrial Informatics (INDIN)* (Bochum, Germany: IEEE) pp 453–457 ISBN 978-1-4799-0752-6
[13] Zhao P, Wang J, Xia J, Dai Y, Sheng Y and Yue J 2012 *Renewable Energy* **43** 234–241 ISSN 09601481
[14] Lima J M, Guetter A K, Freitas S R, Panetta J and de Mattos J G Z 2017 *J Control Autom Electr Syst* **28** 679–691 ISSN 2195-3880, 2195-3899 number: 5
[15] Chen N, Qian Z, Nabney I T and Meng X 2014 *IEEE Trans. Power Syst.* **29** 656–665 ISSN 0885-8950, 1558-0679 number: 2
[16] Hoolohan V, Tomlin A S and Cockerill T 2018 *Renewable Energy* **126** 1043–1054 ISSN 09601481
[17] Eseye A T, Zhang J, Zheng D, Ma H and Jingfu G 2017 *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)(* (Beijing, China: IEEE) pp 552–556 ISBN 978-1-5090-3618-9 978-1-5090-3619-6
[18] Donadio L, Fang J and Porté-Agel F 2021 *Energies* **14** 338 ISSN 1996-1073 number: 2
[19] Felder M, Kaifel A and Graves A 2010 *Poster presentation Gehalten auf der European wind energy conference*
[20] López E, Valle C, Allende H, Gil E and Madsen H 2018 *Energies* **11** 526 ISSN 1996-1073 number: 3
[21] Fu Y, Hu W, Tang M, Yu R and Liu B 2018 *2018 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)* (Kota Kinabalu: IEEE) pp 217–222 ISBN 978-1-5386-5685-3 978-1-5386-5686-0
[22] Wu Y, Wu Q and Zhu J 2019 *IET Renewable Power Generation* **13** 2062–2069 ISSN 1752-1416, 1752-1424 number: 12
[23] Zhang J, Yan J, Infield D, Liu Y and Lien F s 2019 *Applied Energy* **241** 229–244 ISSN 03062619
[24] Salazar A A, Che Y, Zheng J and Xiao F 2022 *Energy Science & Engineering* **10** 2561–2575 ISSN 2050-0505, 2050-0505 number: 7
[25] Zhang H, Yan J, Liu Y, Gao Y, Han S and Li L 2021 *IEEE Trans. Sustain. Energy* **12** 2205–2218 ISSN 1949-3029, 1949-3037 number: 4
[26] Michalakes J, Dudhia J, Gill D, Henderson T, Klemp J, Skamarock W and Wang W 2005 *Use of High Performance Computing in Meteorology* (Reading, UK: WORLD SCIENTIFIC) pp 156–168 ISBN 978-981-256-354-5 978-981-270-183-1
[27] Volker P J H, Badger J, Hahmann A N and Ott S 2015 *Geoscientific Model Development* **8** 3715–3731 ISSN 1991-9603
[28] Freund Y and Schapire R E 1997 *Journal of computer and system sciences* **55** 119–139 number: 1 Publisher: Elsevier
[29] Hochreiter S and Schmidhuber J 1997 *Neural computation* **9** 1735–1780 number: 8 Publisher: MIT press
[30] Graves A, Fernández S and Schmidhuber J 2005 *Artificial Neural Networks: Formal Models and Their Applications–ICANN 2005: 15th International Conference, Warsaw, Poland, September 11-15, 2005. Proceedings, Part II 15* (Springer) pp 799–804
[31] Goodfellow I, Bengio Y and Courville A 2016 *Deep Learning* (MIT Press)
[32] Chollet F and others 2015 Keras URL `https://keras.io`
[33] Chen Z and Liu B 2018 *Continual Learning and Catastrophic Forgetting* (Cham: Springer International Publishing) pp 55–75 ISBN 978-3-031-01581-6
[34] Iman M, Arabnia H R and Rasheed K 2023 *Technologies* **11** 40 number: 2 Publisher: MDPI