Meteorol. Z. (Contrib. Atm. Sci.), PrePub DOI 10.1127/metz/2020/1005

SPS

Feature selection, ensemble learning, and artificial neural Networks for Short-Range Wind Speed Forecasts

PETRINA PAPAZEK[1*], IRENE SCHICKER[1], CLAUDIA PLANT[2,3], ALEXANDER KANN[1] and YONG WANG[1]

[1]Department of Forecasting Models, Zentralanstalt für Meteorologie und Geodynamik (ZAMG), Vienna, Austria
[2]Faculty of Computer Science, Data Mining, University of Vienna, Vienna, Austria
[3]ds:UniVie, University of Vienna, Vienna, Austria

(Manuscript received October 17, 2019; in revised form January 10, 2020; accepted February 3, 2020)

Abstract

The objective of this study is to provide reliable nowcasting (up to six hours) to short-range wind speed forecasts of up to 40 hours ahead in 10 meters height for meteorological observation sites (i.e., point forecasting). The proposed method is a data-driven approach combining artificial neural networks, ensemble learning, and feature selection techniques. Particularly, we improve a pre-defined baseline setup using meteorological features, pre-classification by forecasting intervals, as well as spatial and temporal related data. This combination of methods is the so-called ZiANN (ZAMG interval artificial neural network) and it is optimized for both nowcasting and short-range forecasts. The developed method is one of the first machine learning based wind speed forecasts for the Austrian domain and Austrian observation sites. Heterogenous data sources are combined to derive training data for ZiANN. In particular, we consider (1) observations from weather stations and (2) output of one or several numerical weather prediction models. For (1), we use data from the TAWES network in Austria, while for (2), we use the AROME, ALARO, and/or ECMWF-IFS model interpolated for the observation site location. The model is validated by two test episodes and selected sites in Austria. Forecasts are compared to alternative methods: a random forest approach, the persistence, the currently operational nowcasting system INCA, the model output statistic META, and the NWP model AROME. Our results show that ZiANN outperforms alternative models, especially in the nowcasting-range. We conclude that machine learning techniques are suitable post-processing tools, which outperform classical methodologies.

Keywords: wind speed, machine learing, feature-selection, ensemble-learning, seamless-prediction, Austria

1 Introduction

Accurate, robust and computationally fast wind speed forecasts are needed for a wide range of applications. Especially sensitive applications such as load balancing (power grid), forecasting for power trading, the estimation of snow accumulation for avalanche services, or optimizing routes in aviation and transport need robust forecasts. Here, forecast frequency and availability can be crucial within a very short time frame, especially for the nowcasting-range (i.e., up to six hours ahead). Currently, weather forecasting is carried out using numerical weather prediction (NWP) models including their underlying physics, often combined for post-processing using classical statistical methods such as model output statistics. Forecasts provided by NWP models can vary in their horizontal and vertical extent and resolution. For instance, the Austrian AROME (SEITY et al., 2011) model covers the Greater Alpine Region (GAR) with a horizontal resolution of 2.5 km. Using such a high resolution guarantees at least to a certain extent the representation of local meteorological phenomena.

However, such resolutions and reproducibility come with high computational demands due to their complex underlying physics and dynamics. Especially for the nowcasting-range, the latency of NWP models with their computational delay of three to four hours past initialization is a drawback. To solve these issues, one can combine the most recent observations with the latest available NWP data. Depending on the complexity of the underlying topography, we require additional post-processing methods. These post-processing methods can include model output statistics or quantile mapping. The aim of these methods is to statistically downscale the coarser NWP model forecast to a higher resolution or even stations or points in a region (GLAHN et al., 1972; PANOFSKY et al., 1968; FEIGENWINTER et al., 2018). Frequently, for nowcasting the persistence model proved to be a skillful benchmark.

More recently, the development of rapid update cycle NWP models, initialized hourly for the next e.g., 12-hours ahead, were developed to overcome the gap in the nowcasting-range. However, also more techniques of the family of artificial intelligence, particularly machine learning, emerged in short-range weather forecasting. Machine learning techniques are generic methods often combining various data sources for training of predictive and descriptive algorithms. In the case of mete-

*Corresponding author: Petrina Papazek,Zentralanstalt für Meteorologie und Geodynamik, Hohe Warte 38, 1190 Vienna, Austria, e-mail: petrina.papazek@zamg.ac.at

orology, one can state that they learn relationships from meteorological observations, spatial dependencies, and NWP model data, depending on the data they are fed with. Once fitted, machine learning algorithms are able to produce post-processed forecasts very fast and often more accurately than other post-processing methodologies. Especially in the field of renewable energy wind speed and power, predictions based on artificial intelligence keep emerging and perform well for different demands.

Ramasamy et al. (2015) used a feed-forward artificial neural network (ANN) for daily wind speed predictions in complex terrain showing that, for daily predictions, an observation-based ANN already provides useful information. Among others, Caam et al. (2005), Ak et al. (2013); Ak et al. (2015), or Pelletier et al. (2016) use similar ANN structures and model input data. Often, machine learning is combined with metaheuristic or pre-processing techniques such as feature engineering, such as by Chang (2013) and Xu et al. (2015). Techniques based solely on NWP data as input can be found, too (Díaz et al., 2015).

Recently, Schicker et al. (2017) showed that combining NWP data and observations can improve the short-range wind speed forecasts when using an artificial neural network. Similar results are found in statistical post-processing of e.g., ensemble forecasts by Delle Monache et al. (2011); Delle Monache et al. (2013); Gneiting et al. (2005). Based on the results of the previous studies and findings, we plan to evaluate the effects of different input data and parameters from NWP models and observations within this study.

While the above-mentioned studies focus solely on ANNs, other studies investigate data pre-processing techniques as an additional step before applying a machine learning algorithm. Kusiak et al. (2009a); Kusiak et al. (2009b) implement a k-nearest neighbor model, combined with a principal component analysis and a filtering algorithm whereas Robert et al. (2013) investigate spatial relationships in the prediction. Particularly, Robert et al. (2013) use a general regression neural network to learn relationships between topographic features, observed monthly wind speeds, and spatial data (e.g.: terrain convexity, terrain height, slope, and exposure from the digital elevation model at different spatial scales). They reveal that learning these spatial relationships between topographic features and wind speed improved the accuracy of their used ANN.

Data mining is the process of discovering patterns in large data-sets involving methods including spatial and temporal relationships. Such data mining techniques substantially improve the skill of our methods, particularly utilizing spatial and temporal relationships between the data sources. Furthermore, using k-means clustering for regime-dependent ANNs could improve forecasts as can be seen in an example for solar radiation (McCandless et al., 2016a,b).

With the availability of more computational power, the usage of complex neural networks emerge in geosciences. In conjunction with machine learning algorithms data mining techniques like clustering and feature engineering are often able to boost a basic machine learning approach. Recent contributions range from different types of recurrent neural networks (RNN), such as the mixture density RNN and an LSTM RNN by Felder et al. (2010), a convolutional LSTM by Shi et al. (2015), and RNNs based on a Nonlinear Autoregressive Neural Network (NAR) by Chatziagorakis et al. (2014); et al. (2016). Long short-term memory (LSTM) describes a group of ANNs with feedback connections able to remembers values over arbitrary time intervals (Hochreiter et al., 1997). Not only different types of neural networks are used but also extreme learning machines (Leuenberger et al., 2015; Laib et al., 2016; Li et al., 2016) and deep learning neural networks (Dalto et al., 2015) are employed. However, a more complex structure such as convolutional neural networks (CNN), imply the usage of larger data-sets.

As NWP models underly frequent change in physics, data assimilation, etc., we intend to use a powerful but feasible feed-forward ANN model and boost its performance by data mining techniques. We develop a data-driven methodology for hourly forecasts of wind speed in ten meters height above ground level using a feed-forward neural network. The network will be tailored to the respective sites for the nowcasting (1–6 hours ahead) and the short-range (up to two days ahead). A basic version of the ANN enables us to evaluate the impact of different data mining techniques. As sources, we consider different NWP and observation data and pre-classify the data to identify relevant features, define spatial relationships, and use temporally related training as well as forecasting episodes. To ensure the robustness of the forecasts, we setup neural network ensembles. The forecasts are validated against observations of the Austrian meteorological observation system (Teilautomatische Wetterstationen, TAWES), the AROME model, a model output statistics (META), and the Austrian nowcasting system INCA (Haiden et al., 2011).

Our methodology represents the first approach for Austria addressing wind speed forecasts by machine learning techniques by utilizing Austrian TAWES data.

The remainder of the paper is organized as follows. In Section 2 the data sources, input as well as comparing models, are described. Section 3 gives an overview of the experiments, the methodology and considered data-driven techniques, and the reference baseline model. In Section 4 we describe the results and in Section 5 we draw the conclusions.

2 Data

The data used in this study are observations from the Austrian TAWES network[1] and NWP forecasts of the Austrian AROME NWP model (Seity et al., 2011). We

[1] https://www.zamg.ac.at/cms/en/climate/meteorological-network (English)

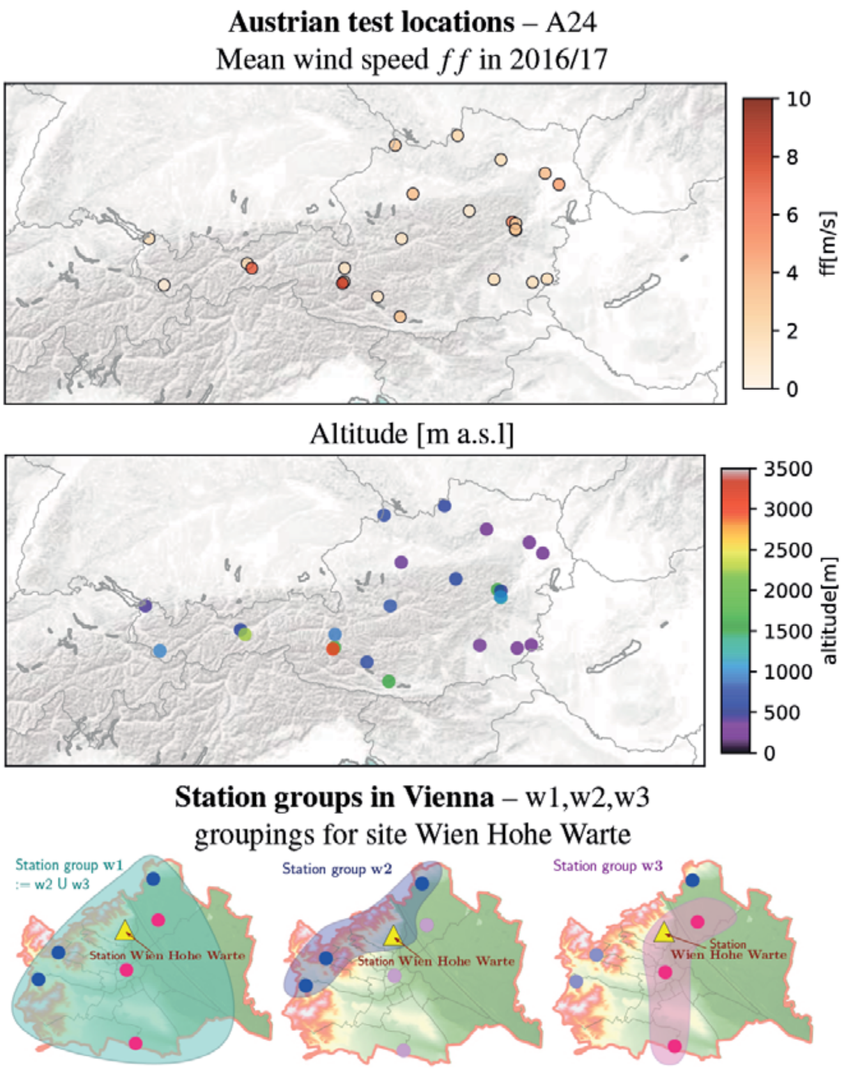

Figure 1: Left: Locations of the selected stations with differing altitudes. Right: Vienna topography and station groups.

select two months, one summer and one winter month (i.e., July 2016 and January 2017) and use a subset of 24 out of the approx. 300 TAWES sites. The selection of the sites was carried out to represent the different Austrian climate zones (Figure 1). To evaluate the skill of the proposed methods, we use data of the INCA model and the model output statistics model META, which combines different sources of global and regional, lagged NWP models.

2.1 Observations

The Austrian TAWES network consists of approx. 300 semiautomatic observation sites unevenly distributed over Austria. They provide meteorological parameters every 10 minutes measured two meters (2 m) or ten meters (10 m) above the ground – e.g., 2 m temperature (T), 10 m wind speed (ff), 10 m wind direction (dd), surface pressure (p), 2 m relative humidity, precipitation, sunshine duration.

The average wind speed for the selected sites is 2.6 m/s with a left-censored distribution and generally prevailing low wind speeds. Higher wind speeds tend to occur in summer more frequently than in winter. The prevailing wind direction for the selected sites is north-westerly. Due to seasonal changes, particularly in the distribution of wind speed and direction, it is meaningful to select training data of the same season as testing data. The characteristics across the sites are substantially different. This issue becomes particularly relevant if artificial intelligence or statistical models integrate several sites at once without any kind of additional pre-processing. Sites located in harsh regions, such as mountain tops, recurrently cause outliers, which need to be fixed, as we further explain in Section 3.

2.2 Numeric Weather Prediction Models

Numeric Weather Prediction (NWP) models provide a large variety of meteorological parameters not only for surface or sub-surface but also in the vertical. Here,

Table 1: Available NWP models.

Model	Prediction horizon & Frequency	horizontal Resolution	Update frequency	Used parameters
AROME *Application of Research toOperations at MEsoscale*	60 hours; hourly	2.5×2.5 km	3-hourly	ff, dd, T, p
ALARO *Aire Limitee Adaptation/Application de la Recherche a l'Operationnel*	72 hours; hourly	4.8×4.8 km	6-hourly	ff, dd, T, p
ECMWF IFS *European Centre for Medium-Range Weather Forecasts Integrated Forecasting System*	> 90 hours;hourly (up to +90 h)	9×9 km	12-hourly	ff, dd, T, p

Met. parameters: ff: wind speed at 10 meters above ground level, dd: wind direction at 10 meters above ground level, T: air temperature (ground level), p: surface air pressure

we retrieve only surface layer fields interpolated bilinear to the TAWES sites. The surface layer fields correspond to meteorological parameter near the surface, particularly temperature measured at 2-meters and wind speed and direction (or their components) measured 10-meters above the ground. We use data of three NWP models, namely the AROME (Seity et al., 2011), the ALARO (Termonia et al., 2018), and the ECMWF IFS (ECMWF, 2016a; ECMWF, 2016b) model (hereafter referred to as the ECWMF model). AROME and ALARO are both regional non-hydrostatic models part of the ALADIN family, whereas ECMWF is a global model. AROME provides the highest resolution in Austria and in the Alpine region with 2.5 km and is a convection permitting model. ALARO will be dismissed by the end of 2019. Depending on the model, the NWP output is available between three to six hours after their initialization due to the NWP's high computational complexity. AROME provides eight forecast runs per day. In Table 1 a summary of the models is given.

2.3 Integrated Nowcasting through Comprehensive Analysis – INCA

The INCA (Integrated Nowcasting through Comprehensive Analysis) model (Haiden et al., 2011) is a dynamical-statistical model providing gridded analyzes and nowcasting fields. The INCA model covers the whole of Austria including parts of neighboring countries with a horizontal spatial resolution of 1 km. This resolution is necessary to resolve the complex topography of the Alps which is a large part of Austria. It is initialized every hour for the next 48-hours ahead.

The INCA system combines all available observation data, weather radar data, topography and background model data (e.g., gridded forecasts for temperature, humidity, wind, precipitation amount, precipitation type, cloudiness, and global radiation) to produce a gridded analysis and forecasts. For the nowcasting-range INCA uses a weighted combination of most recent observations and the model led trend of the AROME model. It employs classical correlation-based motion vectors derived from previous consecutive analyses. After 2–6 forecast hours the nowcast is merged into an NWP forecast, such as AROME. INCA provides wind speed forecasts in-between stations, however, we solely employ forecasts interpolated to the selected sites for our evaluation. More detail about INCA can be found in (Haiden et al., 2011).

2.4 Model Output Statistics (MOS)/META

The META model output statistics uses the information of available global and regional NWP models including lagged versions. It provides weighted forecasts according to the skill of the individual NWP models of the recent past. Recent changes are included via the model forecast bias of the latest available observations. Similar to INCA, META processes the wind speed forecasts.

3 Methodology

In Figure 2 an overview of the methodology is given. Pre-processing and data analysis techniques are applied to the input data in order to perform feature selection.

3.1 Description of the experiments

In total ca. 350 different settings and test scenarios were carried out (see Table 2 for a summary). Additionally, we define two different groups of observation sites. The first consists of the 24 selected TAWES sites (A24, see Figure 1) whereas the second focuses on Vienna and its seven sites with the main focus on the site Wien Hohe Warte (Figure 1).

Forecasts are validated for two episodes, July 2016 and January 2017, using the average performance and extreme events (not discussed here). Hourly forecasts for the next 40 hours ahead are produced for those two months.

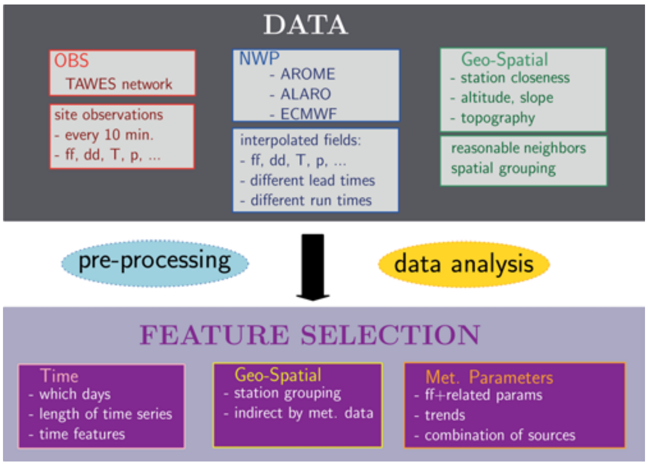

Figure 2: Overview on the methodology: data sources and feature selection.

3.2 Feature selection and preprocessing

Features describe the input or output of a machine learning model. Here, these features are based on observations and NWP model data (Sect. 2). Wind speed measured 10 meters above the ground is the target parameter (denoted henceforth as ff). Wind speed is a continuous, however also a highly variable meteorological quantity. It serves as the single output feature of all experiments.

Wind speed and direction is driven by many parameters such as topography, sunshine, surface pressure, local and large scale pressure differences, surface roughness, turbulence, and temperature. A careful selection of input data is needed to be able to reproduce past observations in the training period. Even more so to perform predictions. Here, observed wind speed (ff), wind direction (dd), 2 m above ground level air temperature (T), surface pressure (p) proved to be useful input features (experiment not shown here). For NWP data the following surface level parameters are used: 2 m temperature (temperature in 2 meters above the ground), wind speed and direction of the 10 m wind (horizontal wind measured at 10 meters above the ground). Additionally, we consider a history of the latest available observations, including their age (e.g., 10-minutes old). The history length of observations is arbitrary and we chose it empirically through experiments. For the feature selection of the NWP models, again, wind speed, wind direction, temperature, and surface pressure are chosen. Missing values in the observation or the NWP data are replaced using linear interpolation unless the data gap is too large (e.g., if there are six missing values in a row, or 1-hour of missing data occur these records are omitted). An in-depth data analysis has shown that the behavior and relationships between the parameters vary to a very large extent among the stations. To better consider the regional differences and the effects of the topography for each site, location-based data-sets for each station or a preselected group of stations (see Section 3.4.2) were generated.

Table 2: Overview on performed experiments, in total ca. 350 (ep. = test episode, n.u. =non-uniform); (x,y) = (steps of ff+dd+T+p, steps with only ff); MSE = mean squared error, MAE = mean absolute error, MAPE = mean absolute percentage error, MSLE = mean squared logarithmic error (see also Appendix A).

Set of experiments	**Tested configurations**	**runs/ep.** (forecasts)
Baseline (preliminary)	different ensemble interval setups, network setups ...	6 (8928)
Intervals and Ensemble	ensemble: 1, 5, 10,15, 20 interval overlap/disjoint intervals: 1-sized, 2-sized, 5 n.u., 6 n.u, 8 n.u	17 (25296)
Net Optimization	1–5 layers; 16–100 Neurons algo: RMSProp, Adam, SGD, Nadam, Adamax, Adagrad obj.: MSE, MAE, MAPE, MSLE	76 (113088)
Time Series Length	full: (1,0), (2,0), (3,0), (4,0), (5,0) partial: (1,1), (2,1) (2,2), (1,3)	8 (11904)
Multi Model Input	1 model: AR (AROME), AL (ALARO), or EC (ECMWF) 2 models: AR+AL, AR+EC, AL+EC 3 models: AR+AL+EC	9 (13392)
OBS Features	4 param.: $ff + dd + T + p$ 3 param.: $ff + T + p$, $ff + dd + T$ 2 param.: $ff + dd$, $ff + T$, $ff + p$ 1 param.: ff	12 (17856)
NWP Features	4 param.:$ff + dd + T + p$ 3 param.: $ff + T + p$, $ff + dd + T$ 2 param.: $ff + dd$, $ff + T$, $ff + p$ 1 param.: ff projected: Δff, ΔT	15 (22320)
Rolling, Fixed, Seasonal Model	rolling horizon: 120 d fixed horizon: 30, 60, 90, 120, 150, 210 d seasonal: 1, 2, 3, 4 years (ca. 3 m/a)	12 (17856)
Spatial Grouping Models	regions.: w1, w2, w3 method: neighborhood, grouping, similarity train days: 60, 120	18 (26784)
RF Model	trees: 10, 50, 100, 150, 300, 500	8 (11904)

To use ANN approaches, one needs to scale the data between [0,1] as the activation functions are defined to work within this range and the data i.e., the parameters can cover different ranges such as wind speed (0 m/s to 35 m/s) and temperature (max. −50 deg C to 45 deg C). This normalization is also needed to unify the impact of the diverse input features, i.e., meteorological parameters, that come in very different ranges. Without normalization, such diverse input features may impair learning their significance for the output. Therefore, all neurons but also all trees in the random forest use normalized data. Here, the min-max normalization is used.

We add time-related features to model temporal relationships in meteorological data which is otherwise lost by the feed-forward ANN approach. Examples of time-related features are the "observation age" and "NWP forecast age". The "observation age" is the time between the predicted time and the time of the observation record. The "NWP forecast age" gives the most recent NWP forecast's offset to the observation. Besides, all input data are pre-processed. For instance, a periodic function transforms the data before fed into an ANN.

In addition to features extracted from the data directly (i.e., observations, NWP forecasts), we examine *projected features* as input features. They represent the changing trend of meteorological parameters. For instance, we compute the absolute difference between adjacent time-steps of the NWP forecasts for a selected meteorological parameter, such as the surface temperature. These adjacent time-steps would be otherwise not considered in a basic ANN architecture.

The features of the baseline model and training episode length are both adjustable, enabling a set of different experiments. We apply a minimum of eight features to train the models using the previous 120 days of data (see the following subsections).

3.3 Baseline model – Random Forests

As an alternative machine learning algorithm to the proposed ANNs, we implemented a Random Forests (RFs) method (Ho, 1995). RFs comprise a set of decision trees for regression or classification. As the RF's output is the mean value of individual tree outputs, an RF can be seen as an ensemble method.

For this study, we configure an RF regressor for the same input and output features as used in the ANN methods, and we train and test the RF with the same data-set.

The modular choice allows us to gain model flexibility. Indeed, we can easily switch between the ANN and the RF approach by activating the model in the configuration.

As the focus of this study is on the neural network approaches, not much model tuning was carried out for the RFs. We investigated different numbers of trees – the final setup comprises 150 trees.

3.4 Baseline Artificial Neural Network – ZiANN

Artificial neural networks (ANNs) are machine learning methods used to recognize patterns and solve regression tasks. A neural network is described by its network architecture, i.e., the number of layers and connections. They are inspired by the biological neural networks constituting brain structures, which involves learning by identifying characteristics from the processed input data without prior knowledge. They are based on a set of connected nodes, the neurons, transmitting a signal in between them in different layers. The layers arrange the

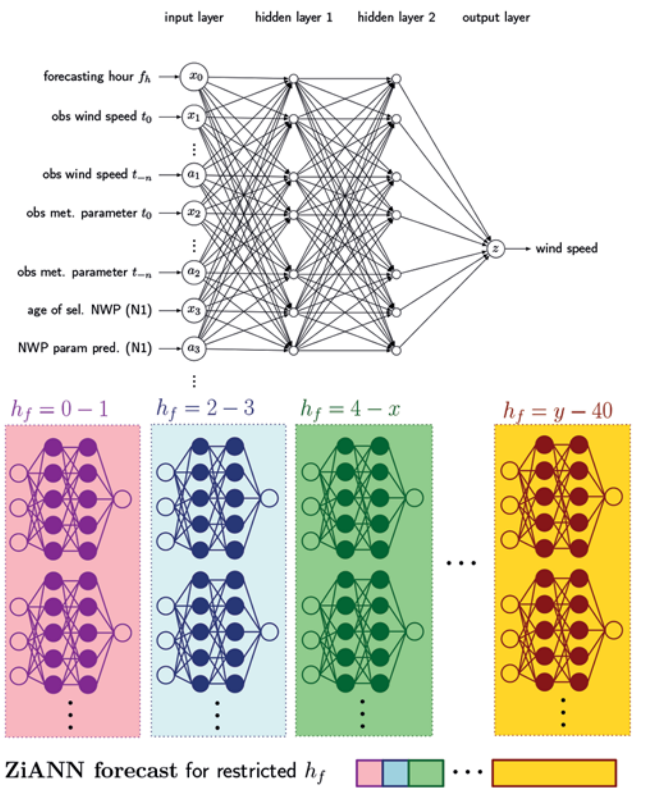

Figure 3: Top: The baseline ANN structure for forecasting on target site; x_i represents normalized input features, a_i additional features (i.e. extending the history), and z the normalized output feature. The length of the input features depends on the configuration of features from observations and NWP models with an 8-dimensional vector as minimal/basic setup. Bottom: The ZiANN model: feed-forward ANNs applied on pre-classified data-sets (intervals of lead times) combined with ensemble learning.

neurons into the input layer, the hidden layer(s), and the output layer. In the simplest case, an ANN consists of one input, one hidden, and one output layer.

In this study, we follow a supervised learning approach, thus we train the model by known target values in the output layer, which improves learning and, hence, the reliability of results. In our applications we define the observed wind speeds 10 meters above the ground as the target values. In particular, a feed-forward ANN (frequently referred to as multiple layer perceptron ANN) is used for all experiments. In Figure 3 the baseline setup of the ANN is shown. Weighted arcs connect neurons, the processing units of the feed-forward ANN, to all neurons of the next layer. Each neuron aggregates input values and generates an output value. An optimization algorithm minimizes the error of the output by adjusting these weights.

Fine-tuning of the neural network was carried out in multiple steps. The input/output features, training data length, and the interaction of various methods were defined in intensive testing phases for all the experiments separately. ANNs address the network optimization problems by gradient descent search algorithms.

Different gradient descent algorithms, such as RMSpop, stochastic gradient descent, and the Adam optimization algorithm (Kingma et al., 2014) were investigated. Results show that the Adam algorithm performs best in conjunction with the mean squared error (MSE) and mean absolute error (MAE) as objective functions for minimization. Experiments on the number of hidden layers reveal that two layers are sufficient for most of the sites. The ANNs run well with medium numbers of neurons, i.e., 50–70. With more neurons, the method tends to overfit.

The ANN architecture itself was defined for all following experiments, based on the ANN baseline version, using the Adam algorithm as an optimizer, MSE as the objective, and two hidden layers with 64 neurons each. As activation functions, we use the *hyperbolic tangent* (tanh) and *rectifier/rectified linear unit* (relu) and a small learning rate.

3.4.1 ZiANN – station based approach

The ZiANN station based approach combines three machine learning methods – feed-forward artificial neural networks (ANNs), ensemble learning, and pre-classification by selected forecasting intervals. Therefore, this approach is denoted as the *ZAMG interval-based ANN ensemble method* (ZiANN, see Figure 3).

For every selected site ZiANN is trained individually using the data specific to the location. The *ensemble learning technique* addresses the robustness of the ZiANN. Ensemble learning refers to training multiple instances of ZiANN, using the same network architecture and training data but applying the randomized initialization (Friedman et al., 2001). Randomly initialized weights serve as a starting point for the optimization process during the training (refined by each iteration of the optimization algorithm). Hence, one can consider ZiANN as a non-deterministic model. For the final forecast evaluation, the ensemble mean is used.

To obtain good results for all forecasting ranges, a *pre-classification by intervals* of the lead time is applied. Individual ZiANN sub-models address different forecast horizon intervals, where an interval's length is independent of the neighbouring intervals. Thus, we remove complex temporal relationships of the data and ease learning.

However, one has to take care not to reduce the amount of data too much and ensure a certain amount of training data samples. Therefore, a minimum training length is needed.

We analyze four possible interval settings (see Figure 4): (1) uniform one sized disjoint intervals, (2) uniform two sized disjoint intervals, (3) non-uniform sized disjoint intervals, and (4) non-uniform sized overlapping intervals. Uniform, i.e., equally sized,intervals indicate that we (1) associate each interval with one lead time or (2) with two lead times (i.e., 40 or 20 intervals). Both apply disjoint intervals, which means to use the

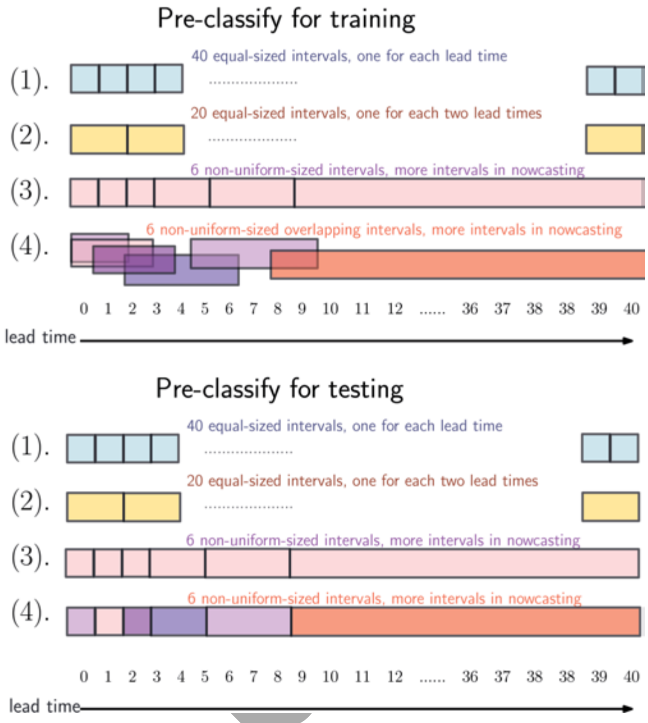

Figure 4: Different methods for pre-classifying by forecast intervals (1, 2, 3, 4) investigated in this study; Top: The pre-classification of the training data set; Bottom: The pre-classification of the testing data set.

same forecasting hour in training and testing data. Typically, relationships in nowcasting are complex and require many intervals in this range. Non-uniform intervals (3) provide this flexibility to define several intervals in the nowcasting-range (defined here as +1–6 hours) and few in later forecasting hours. In (4), we again use non-uniform intervals. However, we overlap neighbouring non-interval bounds for the training to increase the amount of training data and variance. The most successful method proved to be the latter, (4) – therefore, only those results are considered in this study.

3.4.2 ZiANN – spatial grouping approach

Taking spatial relationships and topography into account generally improve forecasts, especially in complex terrain. For instance, considering northwesterly flows, data of a site upstream can provide useful information. Furthermore, similarities between observation sites, such as those along the Danube River, can be exploited to learn relationships and increase the data-set length.

Therefore, we investigate spatial station grouping methods: the (8) *neighborhood*, the (9) *grouping*, and the (10) *similarity* method. The grouping of sites implicitly enables ZiANN to consider topography and other relevant spatial features. Thereby, forecasts for mountainous sites and sites which are challenging for machine learning methods can significantly improve. For this study, a spatially interesting group was selected manually based on knowledge of the research team.

We plan to replace this method in the future by a novel unsupervised clustering method (student work by C. Pacher, 2019, publication to appear).

For the *neighborhood method* (8) ZiANN learns from spatial neighbor stations complementing the target station. The model receives additional input features to integrate information from these neighbors. We select a small group of stations for a target station. The *grouping method* (9) uses data of several sites in a (possibly larger) station group for the training. Here, ZiANN's training data is composed of joined data of all stations. Still, it applies the same model to forecast within the chosen group. Thereby, the variability and amount of data in the training data-set increases. However, this requires stations within a station group to have similar behavior/characteristics. The *similarity method* (10) combines both approaches to address sites with too small training data-sets. Training data using their spatial neighbors extend the pool of available data. This training data utilizes the neighbooring station's input features but the target stations output values for generating the training data set. Thus, (10) exploits the relationship and similarity of neighboring observation sites.

3.4.3 ZiANN – time horizon experiments

Given the nature of meteorological data, ZiANN learns using recorded observations and past NWP forecasts. For instance, for hourly issued two-day-ahead forecasts, training data with at least 30 days of past observations are needed (see Section 4). We consider different sets of training data starting from 30 past days up to 210 days. Extending training data generally improves the forecast skill unless it differs too much from the testing data. As data analysis indicates to focus on the same season, i.e., periods of similar weather situations, we consider different temporal horizons.

We use the three following methods: the (11) *rolling horizon*, the (12) *fixed horizon*, and the (13) *seasonal method*. For (11), we roll the time horizon of training data each day, which results in retraining ZiANN for every forecast day. For (12), we fix the training data for several forecasts. We use, for example, the same model to forecast for a whole month without retraining. For (13), we use years of data where we omit other seasons. With seasonal data, we fill the data pool by more useful situations for the forecast period. In experiments, the latter works best proving that the season is relevant for weather-related parameters. Method (12) works equally well as (11) and we, thus, prefer (12) hereafter for saving computational resources.

4 Results and discussion

In this study, in total ca. 350 experiments (see Table 2) were performed focusing on three main topics: (I) reliable nowcasting and short-range forecasts of wind speed using machine learning, (II) implement a data-driven ensemble approach, and (III) apply spatial and temporal data mining techniques to define the best forecasting setup.

4.1 Feature selection results

Different meteorological parameters were investigated. However, we reveal that for wind speed forecasts, the parameters temperature (T), air pressure (p), wind direction (dd), and wind speed (ff) at the surface level perform best. Other parameters did not improve the forecasts to an extent that the extra computational costs would be justified. Furthermore, it turned out beneficial to use the most recent with a history of past measurements. Still, the most important parameter is the most recent observed wind speed ff (see Figure 5). However, using all parameters improves the forecasts even more.

Investigating the number of past time-steps of the observed parameters showed that four past time-steps improved the nowcasting already sufficiently enough even though up to ten were used (not shown here). However, keeping data transfer of various observations sources in mind, e.g., for wind energy applications, often one does not get the past ten observations within a reasonable amount of transfer time. Therefore, using four past time-steps, i.e., the past 30 minutes, is sufficient enough to cover recent trends.

Having a portfolio of at least three NWP models we evaluated if using data of more than one NWP model improves the forecast skills of the proposed baseline model (see Section 4.2). ZiANN performs best for the AROME, the NWP with the highest spatial resolution. The AROME model provides an average mean absolute error (MAE) of 1.40 m/s in the investigated test episodes. However, also ECMWF and ALARO show a good forecast performance with an MAE of 1.68 m/s and 1.30 m/s. This implies that in the case of not having a convection-permitting model such as AROME, the usage of the ECMWF model would be a good choice. The ECMWF model, with its reduced variance due to the coarser topography, most likely might simplify the learning process as the fluctuations are reduced in contrast to the other models. Thus, using less training data might be sufficient for the ECMWF model. However, ECMWF appears to be redundant if AROME is available in most of our study cases. Investigating a combination of the three NWP models did not show any further improvements. Table 3 shows the resulting scores (description in Appendix A) of the NWP models and INCA, an alternative statistical model.

4.2 ZiANN – baseline model results

Results of the baseline ZiANN are evaluated against observations, NWP model forecasts of ALARO, AROME, and ECWMF, and statistic-dynamical model forecasts of INCA and META. Summarizing the setting of the baseline ZiANN: as input, the past two available observed time-steps – as well as – NWP model surface level forecasts of wind speed, wind direction, temperature and pressure for the respective lead times are used. The training data comprises 120 days before the first forecasting

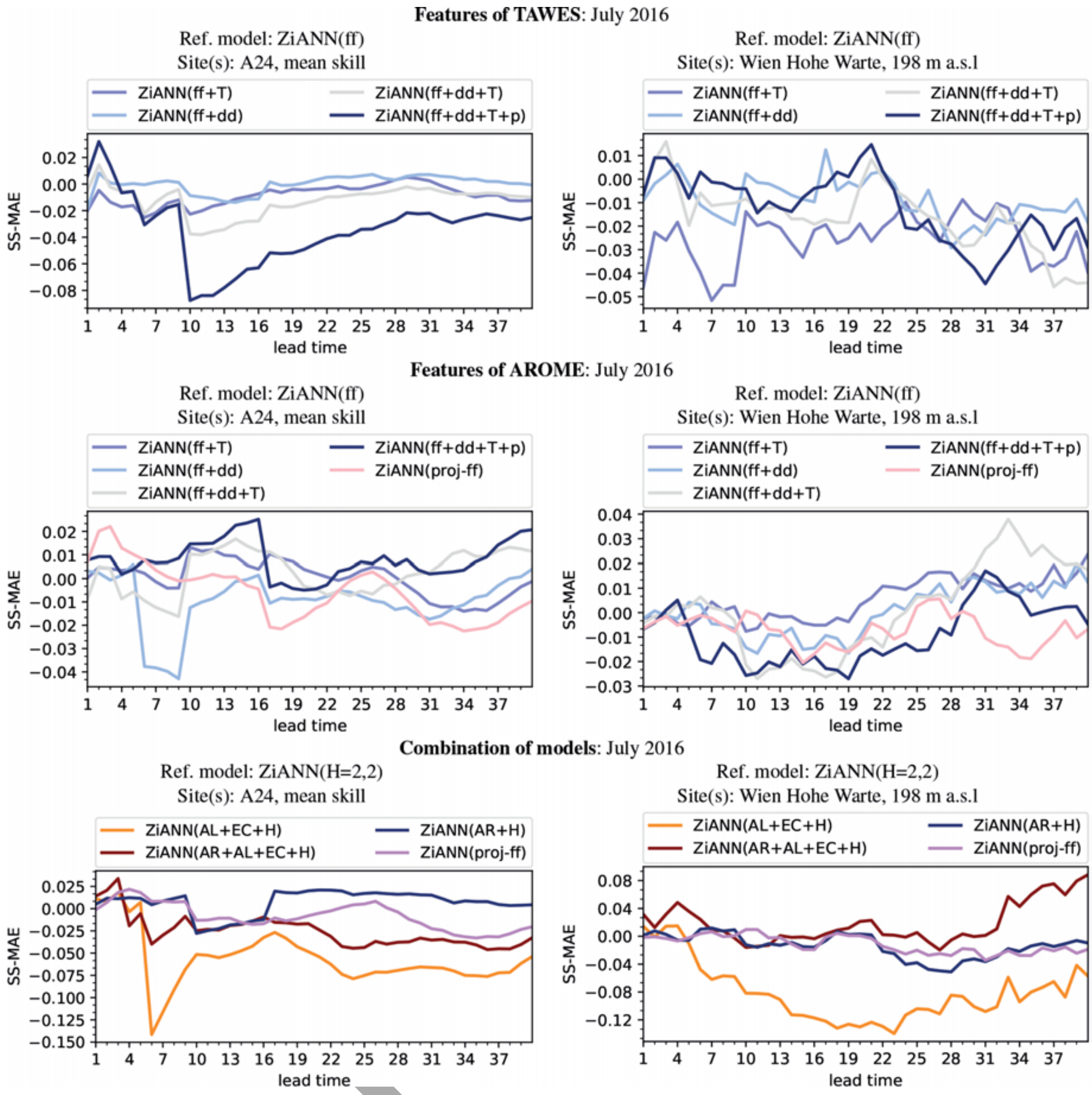

Figure 5: Selected experiments showing the influence of the meteorological parameters on the ZiANN forecasts for the skill-score of the mean absolute error (SS-MAE) in July 2016. Top: effect of using different features ff, dd, T, p from the observations; center: effect of using different features ff, dd, T, p of the NWP model AROME. The experiment "proj.-ff/-T" includes all used meteorological features plus the wind speed/temperature trend. Bottom: using different NWP models and different lengths of time series of the observation(AR:AROME, EC:ECMWF, AL:ALARO, H=2,2: two historic time-steps with all parameters ff+dd+T+p, two further time-steps with the ff parameter only). The left column shows results for the 24 selected stations in Austria, A24, whereas the right column shows the station Wien Hohe Warte. As reference model (referred to as "Ref. model") we use ZiANN with a basic setting (see SS-MAE plot caption).

day. We train the baseline ZiANN with a single forecast interval (i.e., no pre-classification) and ten ensemble members.

Evaluations of the results show that ZiANN yields an overall good performance for the two selected episodes with mean absolute error (MAE) of 1.09 m/s and root mean squared error (RMSE) of 1.42 m/s(baseline version). However, INCA and the persistence model still outperform the ANN for the first two forecasting hours, the so-called nowcasting-range, with MAE 0.97 m/s (ZiANN-120 d) by a reduced MAE of 0.95 m/s (INCA) and 0.87 m/s (persistence). Results of urban stations, such as *Wien Hohe Warte*, imply that ZiANN is still superior to NWPs within the first couple of hours (see Table 5). For the remaining forecasting hours, ZiANN is close to the used NWP. ZiANN generally outperforms other methods in mountainous terrain like Sonnblick. These results imply that one has to use a different strategy, especially for the nowcasting-range, to be able to outperform the persistence model and the two statistical methods.

In contrast to ZiANN, the RF implementation (using 150 trees and the same data-set as input) performs usually very good in the first forecasting hours – it achieves an MAE of 0.90 m/s in the first six forecasting hours whereas ZiANN (final configuration: 120 d) per-

Table 3: Overall scores for July 2016: mae = mean absolute error, corr = correlation, std = standard deviation for a subset of forecasting hours. Top: Alternative models such as NWPs and the statistical model INCA; bottom: different setups of ZiANN (ZiANN-BL = baseline version, see Section 4.2; ZiANN-120d = fixed horizon method with 120 days of training, see Section 4.3; ZiANN-15m = seasonal method with 15 months of data in 2013–2016, see Section 4.5) and the RF model configured with the same data-set and 150 trees.

Alternative models

	INCA			**ALARO**			**AROME**			**ECMWF**		
f_h	***corr***	***mae***	***std***	***corr***	***mae***	***std***	***corr***	***mae***	***std***	***corr***	***mae***	***std***
	0.62	**0.89**	1.47	0.44	1.26	1.00	0.46	1.37	1.42	0.11	1.69	1.00
2	**0.53**	**1.03**	1.50	0.43	1.27	1.01	0.46	1.37	1.42	0.11	1.69	1.01
3	**0.49**	**1.10**	1.45	0.43	1.27	1.02	0.46	1.36	1.42	0.11	1.69	1.02
4	**0.48**	**1.16**	1.45	0.43	1.28	1.02	0.46	1.36	1.43	0.11	1.69	1.02
5	**0.46**	**1.24**	1.50	0.43	1.28	1.02	0.46	1.36	1.43	0.11	1.70	1.02
6	0.44	1.35	1.59	0.43	**1.29**	1.02	**0.46**	1.36	1.43	0.11	1.70	1.02
9	0.43	1.36	1.60	0.42	**1.30**	1.01	**0.45**	1.38	1.45	0.12	1.70	1.01
12	0.43	1.37	1.61	0.42	**1.30**	0.98	**0.45**	1.39	1.46	0.13	1.68	0.98
18	0.43	1.37	1.58	0.42	**1.31**	0.96	**0.43**	1.42	1.43	0.14	1.67	0.96
24	0.42	1.37	1.57	0.41	**1.31**	0.98	**0.44**	1.40	1.41	0.13	1.69	0.98
36	0.40	1.40	1.60	0.39	**1.32**	0.98	**0.42**	1.42	1.43	0.15	1.68	0.98
40	0.40	1.40	1.60	0.40	**1.31**	0.97	**0.41**	1.43	1.42	0.13	1.68	0.97
$\bar{f}_h$	0.43	1.34	1.58	0.41	**1.30**	0.98	**0.43**	1.40	1.43	0.13	1.68	0.98

Proposed models

	ZiANN-BL			**ZiANN-120d** (12)			**ZiANN-15m** (13)			**RF-150trees**		
f_h	***corr***	***mae***	***std***	***corr***	***mae***	***std***	***corr***	***mae***	***std***	***corr***	***mae***	***std***
1	0.54	0.96	0.95	0.65	0.77	1.13	0.63	0.79	0.97	**0.66**	**0.74**	1.15
2	0.52	0.98	0.94	0.58	0.86	0.99	0.57	0.83	0.93	**0.58**	**0.83**	1.08
3	0.50	0.99	0.93	**0.55**	0.90	0.97	0.54	**0.86**	0.93	0.53	0.89	1.07
4	0.49	1.01	0.93	0.53	0.92	0.93	**0.54**	**0.89**	0.94	0.49	0.94	1.08
5	0.48	1.02	0.92	**0.52**	0.94	0.93	0.52	**0.91**	0.90	0.48	0.96	1.08
6	0.47	1.03	0.92	0.51	0.94	0.89	**0.52**	**0.92**	0.91	0.46	1.02	1.14
9	0.44	1.07	0.91	**0.49**	0.97	0.88	0.49	**0.95**	0.86	0.43	1.05	1.18
12	0.44	1.08	0.89	**0.48**	0.99	0.94	0.46	**0.96**	0.85	0.43	1.14	1.34
18	0.41	1.11	0.86	0.45	1.02	0.95	**0.45**	**0.99**	0.89	0.40	1.12	1.09
24	0.40	1.09	0.83	**0.47**	1.00	0.91	0.44	**0.97**	0.82	0.40	1.11	1.09
36	0.38	1.13	0.81	**0.46**	**1.01**	0.86	0.40	1.02	0.76	0.38	1.14	1.12
40	0.36	1.14	0.81	**0.44**	**1.01**	0.85	0.38	1.05	0.76	0.36	1.20	1.16
$\bar{f}_h$	0.42	1.09	0.86	**0.48**	0.98	0.91	0.45	**0.97**	0.83	0.42	1.09	1.14

forms with 0.89 m/s and the persistence with 1.10 m/s. The ZiANN-baseline (i.e., without using intervals in the nowcasting-range) performs with MAE 1.00 m/s. However, for later forecasting RF still yields good results with an average MAE of 1.06 m/s but ZiANN (final configuration) is still better which achieves 0.97 m/s.

Case studies also indicate that ZiANN handles complex situations like mountain tops or events better than the RF (e.g., MAE of 2.03 m/s by the RF and 1.99 m/s by ZiANN at the Sonnblick mountain observatory). This indicates that ZiANN is the overall best choice for our application. Table 3 and Figure 8 show the performance of the baseline model along with the input models and the final variant for the selected stations A24. The, sometimes, sharp bend between lead time 17 and 18 originates in the specification of the defined forecasting intervals (see Section 3, Figure 4). This will be changed in a follow up version using a smoother transition between the intervals.

4.3 ZiANN – station based interval approach

In these series of experiments ZiANN is trained separately for every defined forecasting time-step, every station and every ensemble member. We examine four approaches of interval splitting, described in Section 3.4.1. Using the smallest possible interval size for uniform intervals, i.e., for a single forecast hour, generally gives the highest scores in improving the results, especially in the nowcasting-range (MAE of 0.89 m/s and RMSE of 1.18 m/s in the forecasting hours 1–6). However, the observations and NWP appear to be more uniformly weighted beyond the nowcasting-range (+6-hours). Therefore, we decided to use a single interval for the lead time ranging from seven to 40 hours ahead. To prevent large jumps between lead times, we overlap the boundaries of the intervals instead of using disjoint ones. Overlapping by one or two hours performs best and improves the forecasts and extends the training data-sets, enabling better learning of the model. Thus, as the best possible interval method, we employ the non-uniform sized overlapping intervals in the final setup.

Preliminary forecast simulations showed that ZiANN is dependent on the way it is initialized. Therefore, this is tackled by applying an ensemble learning technique using multiple realizations of the forecast and ultimately the ensemble's mean as deterministic forecast (see Sect. 3.4.1). We determine the number of ensemble members in a multi-step approach. Our findings indicate that, generally, using a higher number of ensemble members performs best in the nowcasting-range, while five members seem to be a sufficiently large enough number for later forecasting hours ($h_f > 6$) in the short-range. The method itself involves longer computational times but improves reliability (see results compared to other methods in Table 3 and Figure 8, ZiANN-BL: the baseline version without pre-classification, ZiANN-120d: best setup of the station based variant using the fixed horizon training method).

4.4 ZiANN – spatial grouping

To consider spatial relations, we investigate three methods of spatial grouping: the (8) *neighborhood*, the (9) *grouping* and (10) *similarity* method (details in Section 3.4.2). For this part a target site was chosen, namely Wien Hohe Warte (located in Vienna). In Vienna, seven observation sites are available. They were grouped, based on e.g., land-use, location, etc. into three different groups: $w1$, $w2$, and $w3$ (Figure 1).

Results indicate that for the (8) *neighborhood method* using data of $w2$ is performing best for the nowcasting-range with an RMSE of <1.21 m/s in contrast to $w1$ and $w3$ with an RMSE of <1.22 m/s and <1.25 m/s, respectively. This can be related to the fact that the sites in $w2$

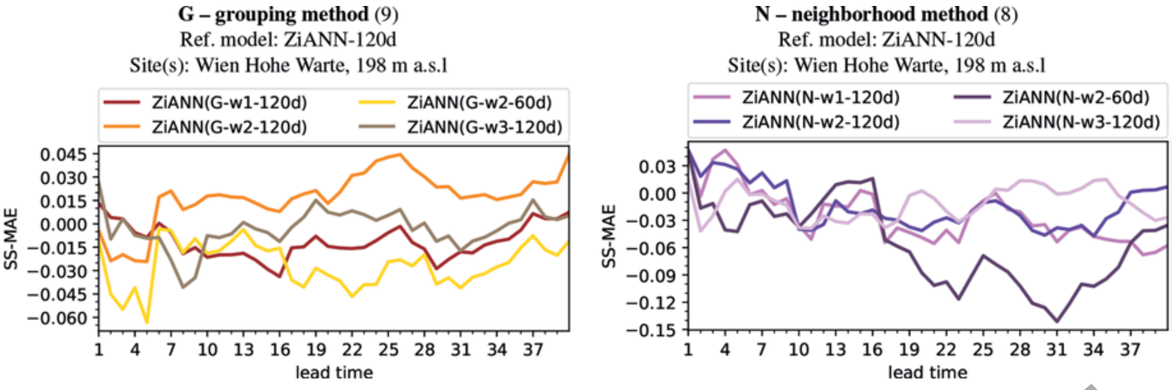

Figure 6: Skill-score for mean absolute error (SS-MAE) for ZiANN experiments for spatial methods (8, 9) for the station Wien Hohe Warte. Shown are results for the *grouping* (left, denoted G) and the *neighborhood* (right, denoted N) method for different lead times. *wi* indicates the cluster, $-xyzT$ the training data length in days and months. As reference model (referred to as "Ref. model") we use ZiANN trained with the previous 120 days of the target site only.

share characteristics of being along the Vienna forest region and more sub-urban than the other sites. Similarly, as all seven Viennese sites are grouped in $w1$ this cluster provides good setup, too. This most likely is due to ZiANN giving more weight to similar sites than to others.

Same conclusions can be drawn for the (9) *grouping method* where the spatial characteristics are equally important. The grouping method yields good forecasts, especially for beyond the nowcasting-range with RMSE of 1.29 m/s (in total). If stations share similarities, it is straight forward to use a grouping.

The (10) *similarity method* is a good method when all other methods fail due to insufficiently long training data. Here, we reduced the training data for Wien Hohe Warte and added spatially close stations – it yields a total RMSE of 1.34 m/s.

Results (Figure 6) show that the neighborhood method performs with MAE 0.99 m/s and the grouping with 0.96 m/s, both being significantly better in the nowcasting-range, where they reach MAE below 0.9 m/s in the first hours.

Results in Table 4 show that using the *neighborhood method* for the nowcasting-range and the *grouping method* for stations which share similar characteristics is the preferred configuration for ZiANN. In complex terrain, the *spatial grouping methods* overpower the single-site data models (see the previous section). However, especially the *neighborhood method* requires a sufficient amount of data for training (i.e., at least 120 days). Thus, it is crucial to take the temporal setup into account.

Station grouping is more or less easy when knowing the sites and having a reduced number of sites. However, for more sites, one might need to use specific clustering methods or group stations based on climatic characteristics.

4.5 ZiANN – time horizon experiments

We investigate different lengths of training data and combinations of seasons and years. In particular, we evaluate three strategies (details in Section3.4.3): the

Table 4: Results for July 2016 of ZiANN for station Wien Hohe Warte for four different experiments, namely 120d: the fixed horizon method (12) with 120 days, 15m: the seasonal method (13) with 15 months, N-w2-120d: the neighborhood method (8) with the w2 cluster and 120 days of training data, G-w2-120d: the grouping method (9) with the w2 cluster and 120 days of training data. Best performing experiment is written in bold font.

	120d (12)			**15 m** (13)			**N-w2-120d** (8)			**G-w2-120d** (9)		
f_h	**mae**	**corr**	**std**	**mae**	**corr**	**std**	**mae**	**corr**	**std**	**mae**	**corr**	**std**
1	0.84	0.77	1.52	0.85	0.76	1.48	**0.80**	**0.79**	1.51	0.84	0.76	1.60
2	0.89	0.76	1.38	0.94	0.73	1.28	**0.88**	**0.78**	1.45	0.92	0.75	1.40
3	0.95	0.73	1.35	**0.92**	0.74	1.27	0.92	**0.77**	1.44	0.97	0.73	1.30
4	0.97	0.72	1.28	0.94	0.72	1.24	**0.94**	**0.75**	1.38	0.99	0.72	1.22
5	0.96	0.74	1.27	0.94	0.73	1.21	**0.94**	**0.75**	1.35	0.99	0.73	1.16
6	0.96	0.74	1.32	**0.93**	0.74	1.22	0.95	**0.74**	1.32	0.94	0.74	1.30
9	0.96	0.71	1.32	0.96	0.71	1.20	**0.95**	**0.74**	1.26	0.95	0.72	1.24
12	0.96	0.72	1.33	0.95	0.72	1.23	1.00	0.71	1.31	**0.95**	**0.73**	1.30
18	1.03	0.67	1.21	**0.99**	0.68	1.20	1.06	0.67	1.23	1.01	**0.68**	1.29
24	1.00	0.66	1.16	0.97	0.68	1.15	1.03	0.67	1.17	**0.96**	**0.69**	1.16
36	1.04	0.67	1.16	1.04	0.67	1.04	1.06	0.66	1.10	**1.02**	**0.69**	1.14
40	1.11	0.64	1.21	1.07	0.65	1.04	1.10	0.65	1.09	**1.06**	**0.66**	1.11
$\bar{f}_h$	0.97	0.69	1.24	0.96	0.71	1.18	0.99	0.70	1.23	**0.96**	**0.71**	1.23

Table 5: MAE in case studies: Wien Hohe Warte (urban), Innsbruck (urban), Sonnblick (mountain observatory), Kolm Saigurn (valley); best setup of ZiANN for the site is chosen (spatial grouping/ seasonal/120 d).

	W. Hohe Warte		**Innsbruck**		**Sonnblick**		**Kolm Saigurn**	
f_h	ZiANN	AR	ZiANN	AR	ZiANN	AR	ZiANN	AR
1	0.80	1.02	0.77	0.93	1.45	3.38	0.59	1.37
2	0.88	1.02	0.77	0.93	1.57	3.34	0.62	1.39
3	0.92	1.04	0.74	0.94	1.74	3.33	0.66	1.40
4	0.94	1.05	0.73	0.95	1.84	3.35	0.71	1.41
5	0.94	1.01	0.74	0.96	1.87	3.33	0.72	1.42
6	0.93	0.98	0.72	0.97	1.78	3.33	0.73	1.43
9	0.95	1.00	0.74	0.99	1.88	3.40	0.75	1.45
12	0.95	0.98	0.75	1.00	2.00	3.41	0.83	1.46
18	0.99	1.03	0.83	0.97	2.02	3.52	0.86	1.44
24	0.96	0.99	0.86	0.96	1.99	3.39	0.82	1.44
36	1.02	1.04	1.22	0.98	2.14	3.32	0.79	1.43
40	1.06	1.08	1.19	0.96	2.18	3.35	0.78	1.43
$\bar{f}_h$	0.97	1.01	0.92	0.97	1.99	3.41	0.79	1.42

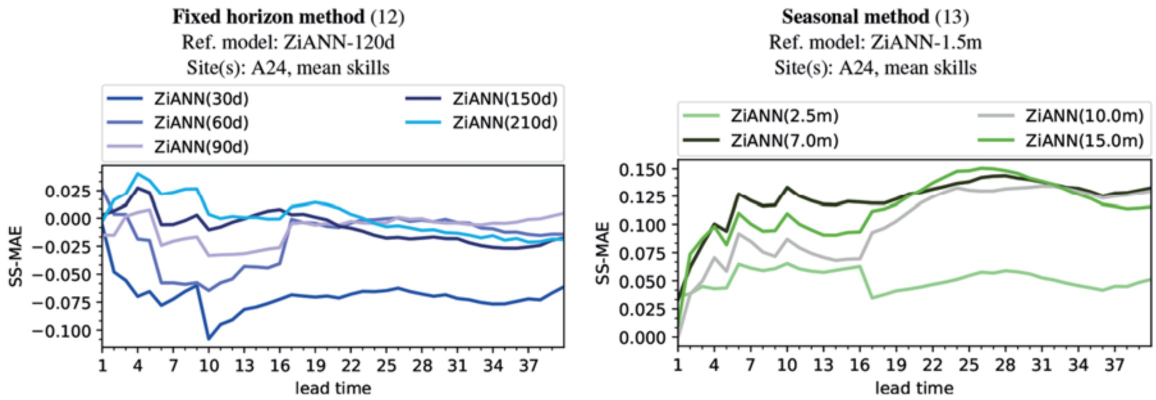

Figure 7: Skill-score for mean absolute error (SS-MAE) for ZiANN experiments of temporal methods (12, 13) – fixed horizon with different training days and seasonal training – for selected stations in Austria (A24) in July 2016: various number of training days (left) for A24; extension by the *seasonal method* (right) for A24 for different lead times. The numbers in parenthesis *xxx*d indicates the number of training days and *yy*m the number of training months. As reference model (referred to as "Ref. model") we use ZiANN trained with the previous 120 days or 1.5 months, respectively.

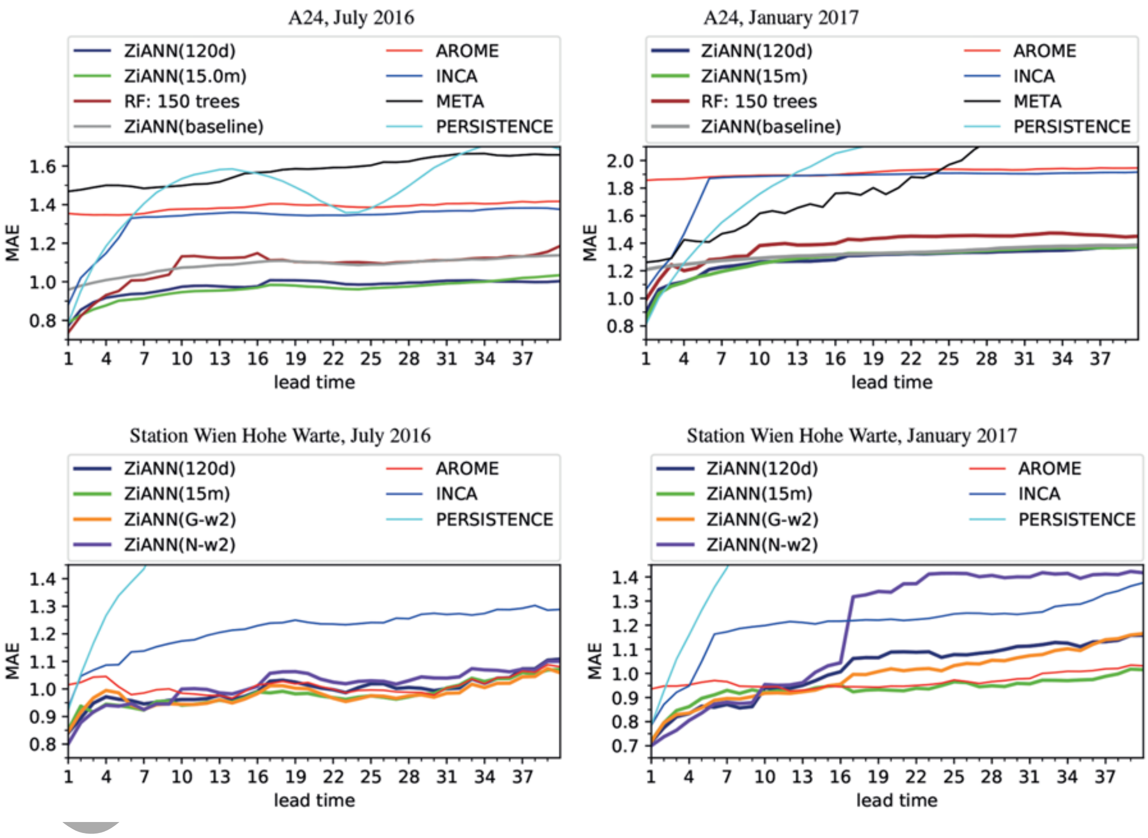

Figure 8: MAE of overall best performing experiments for different lead times for (left) July 2016 and (right) January 2017. Top for the selected 24 Austrian station and (bottom) the station Wien Hohe Warte. If the skill of an alternative model (e.g.: the PERSISTENCE or META model) deviates to a very large instance from our methods we limit the visualized bounds of the y-axis; used abbreviations: "d" = days, "m" = months.

(11) *rolling horizon*, the (12) *fixed horizon method*, and the (13) *seasonal method*.

The *rolling time horizon method* showed a good performance (not shown here). Evaluating methods (11) and (12) show that both strategies scale equally well when using 120 days for model training. This indicates that the data similarity has more influence than the data record time being close to the prediction date. Based on an internal study on wind turbine data a retraining every month or two is advisable. Furthermore, method (11) has a disadvantage compared to the other methods as it requires to train the model for every initialized forecast run. Thus, in operational environments, it would cost additional computational overhead. Therefore, for this

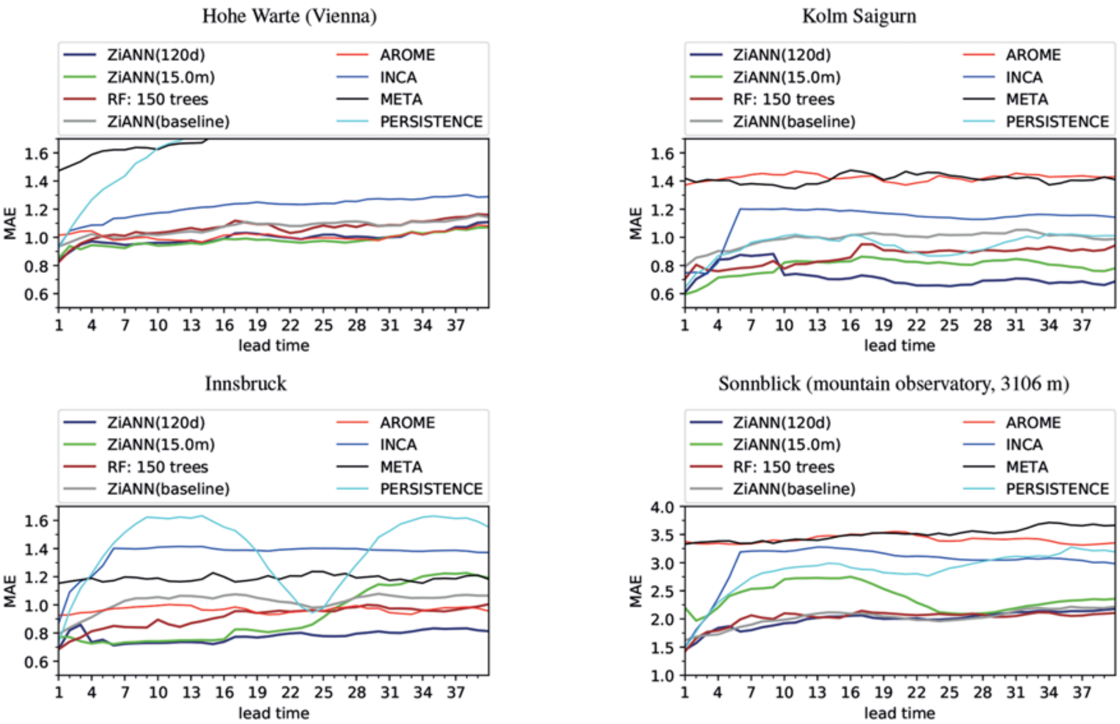

Figure 9: MAE for four selected stations for July 2016: (top left) Wien Hohe Warte (urban area), (bottom left) Innsbruck (urban area), (bottom right) Sonnblick (a mountain observatory), and (top right) Kolm Saigurn (valley station next to Sonnblick observatory). If the skill of an alternative model (e.g.: the PERSISTENCE or META model) deviates to a very large instance from our methods we limit the visualized bounds of the y-axis; used abbreviations: “d” = days, “m” = months.

study, methods (12) and (13) were further investigated. Consequently, we apply the fixed horizon for all subsequent experiments using a fixed number of days before the first day of the forecasting episode to train the model.

Results show that for (12) the best performance was achieved using 120 days for training ZiANN as longer periods impose more seasonal changes (see Figure 7). Through empirically testing the 120 days of training provided the best performing forecasts for our test episodes although it slightly exceeds a season. Generally, the number of days depends, too, on the availability of archived NWP data and data complexity. For example, ZiANN learns to reproduce convective events if the data includes them. Similar to statistical models, if such information is not present in the training data a machine learning algorithm is not able to forecast such events. Furthermore, such relationships are complex to learn and require sufficient data.

Method (13) omits other seasons by concatenating similar seasons (13) from multiple years. It is able to outperform (12) and yields good results for complex situations. Figure 7 (right) gives an example of the result of extended training data for the selected Austrian stations A24. Figure 7 shows different settings of the *fixed horizon* and *seasonal method*. Table 3 includes the results of the best setup using the *seasonal method* in column ZiANN-15 m and results on the *fixed horizon method* in column ZiANN-120d. In most cases the *seasonal method* outperforms the *fixed horizon method* of ZiANN. We obtain an MAE of 0.97 m/s and an RMSE of 1.31 m/s. In the nowcasting-range (1–6 hours) the MAE is reduced to 0.87 m/s and RMSE to 1.18 m/s.

4.6 Overall results

For the overall evaluation, we compare metrics of the best-performing experiments of the afore-described sub-topics against the baseline model, the statistical-dynamical models INCA and META, the raw NWP forecasts, and the implemented random forest (RF) model (Figure 8, Table 3).

For the nowcasting-range the random forest implementation does provide good, sometimes even better forecasts compared to ZiANN. However, in the later lead times, it is not able to catch up with ZiANN. Especially for more complex sites such as mountain sites, the random forest method would need more tuning. As we aim to improve the forecast skill for all lead times and effectively tuned ANNs as base models, we stick to ANNs. Thus, we use the proposed ZiANN model in different configuration for all following experiments. Results for July 2016 for selected sites (see Figure 9) also show that the proposed method ZiANN is able to outperform the other methods for different topographic settings. Figure 10 shows the distribution of the MAE of the ZiANN-120d experiment between stations, in total and the nowcasting range. One can observe that at mountain tops (i.e., difficult situations) the forecasting quality decreases.

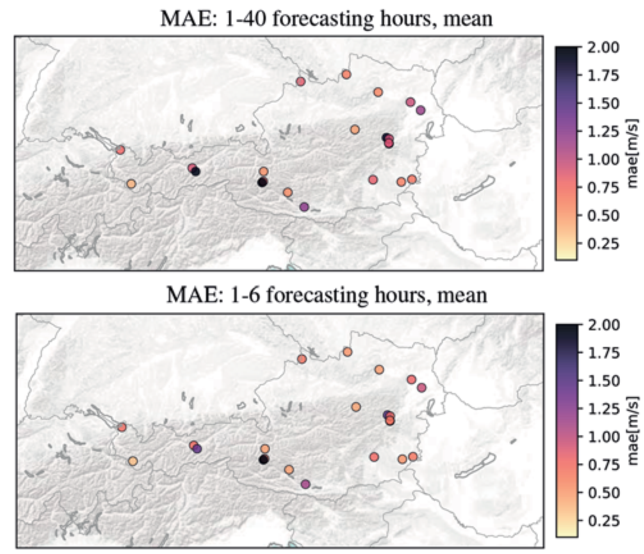

Figure 10: Mean MAE for test cases spatial distribution: (top) 1–40 hours, (bottom) 1–6 hours.

5 Conclusions and Outlook

This study focusses on the development of an optimized setup for a machine learning based nowcasting and short-range forecasting model for wind speed. The idea is to develop a system using observations and NWP forecasts and combine them in a meaningful way using an artificial neural network (ANN). The implemented ZAMG interval Artificial Neural Network (ZiANN) includes ensemble learning techniques, feature selection and considers spatial and temporal relations within the data. Different experiments focusing on training length, the grouping of several sites, and other more data mining related tests have been carried out using two target months, July 2016 and January 2017.

ANNs, ensemble learning, and feature selection techniques are suitable for forecasting wind speeds. In this study, our proposed method includes location-based information and takes local data for each considered site into account. Wind speed-related meteorological parameters (e.g., wind speed, temperature) act as input features of the model. The skill of the model improves when extending both the training lengths of the neural network as well as the amount of most recent observations. The latter is particularly important for nowcasting. For the range beyond the nowcasting-range information of the most recent NWP forecasts is used. Models with more input features, no matter if temporal, spatial or meteorological data, are more complex and cause more computational costs. This can result in the need for more computational resources without necessarily improving the forecast skill, sometimes even overfitting, i.e., negative effects. Likewise, using spatially high resolved NWP model data needs more training data to enable the neural network to learn the characteristics of the NWP model. Therefore, one should select features carefully.

Results of the experiments show that the final setup of ZiANN outperforms the raw NWP forecasts and also statistical-dynamical post-processing methods such as model output statistics. For the nowcasting-range, the random forest model proved to be an interesting alternative or additional model for a sort of probabilistic nowcasting. We have to admit, though, that besides the evaluation of the number of trees no additional hyper-parameter tuning was carried out for the random forest model.

Feature selection and considering the spatial and temporal relations proved to be beneficial for the results of ZiANN. For all forecasting ranges, pre-classifying the data-set by forecasting intervals was important. Furthermore, using small forecasting lead-time intervals in the nowcasting-range solved the issues of giving too much weight to later lead-times in ZiANN. Likewise, the overlapping intervals, i.e., for lead-time two with an interval of ± 1 hour, thus, including information of lead-time one and three, even improved the results. Additional benefits are that one is able to cover time-shifts in the NWP model as well as increasing the training data-set.

We significantly outperform other methods by the station-based approach. ZiANN yields an MAE of 0.97 m/s where INCA gives 1.34 m/s and AROME 1.40 m/s on average for the investigated test episode and location. Even the RF implementation provides an MAE of 1.06 m/s. In the nowcasting-range, we obtain an MAE smaller than 0.95 m/s. By the neigborhood method, we surpass the baseline version of the test site of MAE 0.93 m/s by MAE 0.90 m/s. The seasonal method boosts the results of ZiANN by 0.92 m/s.

This proves that the method is well suited for our test region and a feasible approach for the nowcasting-range, too.

In this study, we showed that basic measures for tuning a neural network model for wind speed forecasting already yields good results. Our method is already part of a semi-operational predicting method in Austria using all TAWES sits and yields a good forecast quality (i.e., outperforming other models such as INCA, AROME, and the persistence). We successfully applied it to data from wind farms in order to give wind speed and power predictions at hub height. Thus, we consider the ZiANN as a robust and efficient forecasting method for the wind speed. However, there are still open topics and issues which need to be tackled to further improve ZiANN. As sites upstream of the current wind direction provide valuable information one could, thus, use a weather dependent grouping or unsupervised clustering method. Also, the tuning of hyper-parameters could still be investigated more deeply by e.g., automatized parameter tuning methods. Limitations of ZiANN are to solely give wind speed predictions for single locations, not for longer prediction horizons than the short-range, relying on sufficient quality checked inputs. In this study, we could only investigate the forecast skill of ZiANN for the described time episodes and set of observation sites.

A Statistical Scores for Deterministic Wind Speed Forecasts

The following scores are used for evaluation, with v_{obs} being the true value of wind speed (ground truth), v_{mod} the predicted value, and n the total number of tested samples:

Mean: a measure for the expected value $E[X]$:

$$\bar{v} = \sum_{i=1}^{n} v_i. \tag{A.1}$$

Mean squared error (MSE): provides the average squared error (i.e., the deviation of the prediction model v_{mod} from the observation v_{obs}) with

$$MSE = \frac{1}{n}\sum (v_{\mathrm{obs}} - \hat{v}_{\mathrm{mod}})^2. \tag{A.2}$$

Root mean squared error (RMSE): square root of the MSE, i.e.:

$$rmse = \sqrt{\frac{1}{n}\sum (v_{\mathrm{obs}} - \hat{v}_{\mathrm{mod}})^2}. \tag{A.3}$$

BIAS: is defined as the difference between the mean of the model and the mean of the true values, i.e.:

$$BIAS = \bar{v}_{\mathrm{mod}} - \bar{v}_{\mathrm{obs}}. \tag{A.4}$$

Mean absolute error: average absolute difference between the model and the true values, i.e.:

$$MAE = \frac{1}{n}\sum |v_{\mathrm{obs}} - \hat{v}_{\mathrm{mod}}|. \tag{A.5}$$

Standard deviation (*STD*): measures the variation of a random variable X and single sample x_i, i.e.:

$$STD(X) = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}. \tag{A.6}$$

We use $STD(v_{\mathrm{mod}})$.

Covariance (*COV*): is the joint variability of two random variables X, Y, to measure whether high values of one variable correspond with high variables of the other variable while low values also correspond,

$$cov(X, Y) = E[(X - E[X])(Y - E[Y])]. \tag{A.7}$$

Here, we evaluate $COV(v_{\mathrm{obs}}, v_{\mathrm{mod}})$.

Pearson correlation coefficient (*CORR*): a normalized version of cov showing the strength of a linear relation, i.e.:

$$CORR = \frac{cov(v_{\mathrm{mod}}, v_{\mathrm{obs}})}{std(v_{\mathrm{mod}})\,std(v_{\mathrm{obs}})}. \tag{A.8}$$

Skill Score (SS): evaluates the forecast v_{mod} against another available reference forecast v_{ref}:

$$SS = \frac{Score(v_{\mathrm{mod}}) - Score(v_{\mathrm{ref}})}{Score(perfect) - Score(v_{\mathrm{ref}})} \tag{A.9}$$

$Score(perfect)$ is the best value of the used score (such as MAE = 0 m/s). $Score(perfect)$ is zero for many scores listed above, which simplifies the computation of SS.

References

Ak, R., Y.F. Li, V. Vitelli, E. Zio, 2013: A genetic algorithm and neural network technique for predicting wind power under uncertainty. – In: Prognostics and System Health Management Conference PHM-2013, 1–6.

Ak, R., V. Vitelli, E. Zio, 2015: An interval-valued neural network approach for uncertainty quantification in short-term wind speed prediction. – IEEE Transactions on Neural Networks and Learning Systems **26**, 2787–2800, DOI: 10.1109/TNNLS.2015.2396933.

Caam, E., E. Arcakliogul, A. Caavusogul, B. Akbiyik, 2005: A classification mechanism for determining average wind speed and power in several regions of turkey using artificial neural networks. – Renewable Energy **30**, 227–239.

Chang, W.-Y., 2013: Short-term wind power forecasting using the enhanced particle swarm optimization based hybrid method. – Energies **6**, 4879–4896.

Chatziagorakis, P., C. Elmasides, G.C. Sirakoulis, I. Karafyllidis, I. Andreadis, N. Georgoulas, D. Giaouris, A.I. Papadopoulos, C. Ziogou, D. Ipsakis, others, 2014: Application of neural networks solar radiation prediction for hybrid renewable energy systems. – In: International Conference on Engineering Applications of Neural Networks, Springer, 133–144.

Chatziagorakis, P., C. Ziogou, C. Elmasides, G.C. Sirakoulis, I. Karafyllidis, I. Andreadis, N. Georgoulas, D. Giaouris, A.I. Papadopoulos, D. Ipsakis, others, 2016: Enhancement of hybrid renewable energy systems control with neural networks applied to weather forecasting: the case of olvio. – Neural Comput. Appl. **27**, 1093–1118.

Dalto, M., J. Matuško, M. Vašak, 2015: Deep neural networks for ultra-short-term wind forecasting. – In: Industrial Technology (ICIT), 2015 IEEE International Conference on, 1657–1663. IEEE.

Delle Monache, L., T. Nipen, Y. Liu, G. Roux, R. Stull, 2011: Kalman filter and analog schemes to postprocess numerical weather predictions. – Mon. Wea. Rev. **139**, 3554–3570, DOI: 10.1175/2011MWR3653.1.

Delle Monache, L., F.A. Eckel, D.L. Rife, B. Nagarajan, K. Searight, 2013: Probabilistic weather prediction with an analog ensemble. – Mon. Wea. Rev. **141**, 3498–3516, DOI: 10.1175/MWR-D-12-00281.1.

Díaz, D., A. Torres, J.R. Dorronsoro, 2015: Deep neural networks for wind energy prediction. – In: I. Rojas, G. Joya, A. Catala (Eds.): Advances in Computational Intelligence, Cham. Springer International Publishing, 430–443.

ECMWF, 2016a: Part III: Dynamics and numerical procedures. – In: IFS Documentation CY41R2, number 3 in IFS Documentation.

ECMWF, 2016b: Part IV: Physical processes. – In: IFS Documentation CY41R2, number 4 in IFS Documentation.

Feigenwinter, S.I., Kotlarski, A. Casanueva, A.M. Fischer, C. Schwierz, M.A. Liniger, 2018: Exploring quantile mapping as a tool to produce user-tailored climate scenarios for Switzerland. – Technical Report MeteoSwiss **11**, 10.

Felder, M., A. Kaifel, A. Graves, 2010: Wind power prediction using mixture density recurrent neural networks. – In: Poster Presentation at European Wind Energy Conference, working paper.

Friedman, T.J. and Hastie, R. Tibshirani, 2001: The elements of statistical learning, volume 1. – Springer series in statistics New York, NY, USA.

Glahn, H.R., D.A. Lowry, 1972: The use of Model Output Statistics (MOS) in objective weather forecasting. – J. Appl. Meteor. **11**, 1203–1211, DOI: 10.1175/1520-0450(1972)011<1203:TUOMOS>2.0.CO;2.

Gneiting, A.E.T. and Raftery, A.H. Westveld, T. Goldman, 2005: Calibrated probabilistic forecasting using ensemble model output statistics and minimum crps estimation. – Mon. Wea. Rev. **133**, 1098–1118, DOI: 10.1175/MWR2904.1.

Haiden, T., A. Kann, C. Wittmann, G. Pistotnik, B. Bica, C. Gruber, 2011: The Integrated Nowcasting through Comprehensive Analysis (INCA) system and its validation over the eastern alpine region. – Wea. Forecast. **26**, 166–183, DOI: 10.1175/2010WAF2222451.1.

Ho, T.K., 1995: Random decision forests. – In: Proceedings of 3rd International Conference on Document Analysis and Recognition, volume 1, 278–282. DOI: 10.1109/ICDAR.1995.598994.

Hochreiter, S., J. Schmidhuber, 1997: Long short-term memory. – Neural Computation **9**, 1735–1780, DOI: 10.1162/neco.1997.9.8.1735.

Kingma, D.P., J. Ba, 2014: Adam: A method for stochastic optimization.

Kusiak, A., H. Zheng, Z. Song, 2009a: Short-term prediction of wind farm power: A data mining approach. – IEEE Transactions on Energy Conversion **24**, 125–136.

Kusiak, A., H. Zheng, Z. Song, 2009b: Wind farm power prediction: a data-mining approach. – Wind Energy **12**, 275–293, DOI: 10.1002/we.295.

Laib, M., M. Kanevski, 2016: Analysis and modelling of extreme wind speed distributions in complex mountainous regions. – In: EGU General Assembly Conference Abstracts, volume 18, 3338.

Leuenberger, M., M. Kanevski, 2015: Extreme learning machines for spatial environmental data. – Comput. Geosci. **85**, 64–73.

Li, Z., L. Ye, Y. Zhao, X. Song, J. Teng, J. Jin, 2016: Short-term wind power prediction based on extreme learning machine with error correction. – Protection and Control of Modern Power Systems **1**, 1–8.

McCandless, T.C., S. Haupt, G. Young, 2016a: A regime-dependent artificial neural network technique for short-range solar irradiance forecasting. – Renew. Energy **89**, 351–359.

McCandless, T.C., G.S. Young, S.E. Haupt, L.M. Hinkelman, 2016b: Regime-dependent short-range solar irradiance forecasting. – J. Appl. Meteor. Climatol. **55**, 1599–1613, DOI: 10.1175/JAMC-D-15-0354.1.

Panofsky, H.A., G.W. Brier, 1968: Some applications of statistics to meteorology. – University Park: Penn. State University, College of Earth and Mineral Sciences.

Pelletier, F., C. Masson, A. Tahan, 2016: Wind turbine power curve modelling using artificial neural network. – Renew. Energy **89**, 207 – 214.

Ramasamy, P., S. Chandel, A.K. Yadav, 2015: Wind speed prediction in the mountainous region of india using an artificial neural network model. – Ren. Energy **80**, 338–347, DOI: 10.1016/j.renene.2015.02.034.

Robert, S., L. Foresti, M. Kanevski, 2013: Spatial prediction of monthly wind speeds in complex terrain with adaptive general regression neural networks. – Int.J. Climatol. **33**, 1793–1804, DOI: 10.1002/joc.3550.

Schicker, I., P. Papazek, A. Kann, Y. Wang, 2017: Short-range wind speed predictions for complex terrain using an interval-artificial neural network. – Energy Procedia **125**, 199 – 206, DOI: 10.1016/j.egypro.2017.08.182.

Seity, Y., P. Brousseau, S. Malardel, G. Hello, P. Bnard, F. Bouttier, C. Lac, V. Masson, 2011: The AROME-france convective-scale operational model. – Mon. Wea. Rev. **139**, 976–991.

Shi, X., Z. Chen, H. Wang, D.Y. Yeung, W.K. Wong, W. Woo, 2015: Convolutional LSTM network: A machine learning approach for precipitation nowcasting. – CoRR **abs/1506.04214**.

Termonia, P., C. Fischer, E. Bazile, F. Bouyssel, R. Brožková, P. Bénard, B. Bochenek, D. Degrauwe, M. Derková, R. El Khatib, R. Hamdi, J. Mašek, P. Pottier, N. Pristov, Y. Seity, P. Smolíková, O. Španiel, M. Tudor, Y. Wang, C. Wittmann, A. Joly, 2018: The ALADIN System and its canonical model configurations AROME CY41T1 and ALARO CY40T1. – Geosci. Model Develop. **11**, 257–281, DOI: 10.5194/gmd-11-257-2018.

Xu, X., D. Niu, t. Fu, H. Xia, H. Wu, 2015: A multi time scale wind power forecasting model of a chaotic echo state network based on a hybrid algorithm of particle swarm optimization and tabu search. – Energies **8**, 12317.