



HAL
open science

Spatio-Temporal Deep Learning for day-ahead wind speed forecasting relying on WRF predictions

E Christoforou, Ioannis Z. Emiris, A Florakis, D Rizou, S Zaharia

► **To cite this version:**

E Christoforou, Ioannis Z. Emiris, A Florakis, D Rizou, S Zaharia. Spatio-Temporal Deep Learning for day-ahead wind speed forecasting relying on WRF predictions. Energy Systems, 2021, 10.1007/s12667-021-00480-6 . hal-03523828

HAL Id: hal-03523828

<https://hal.science/hal-03523828v1>

Submitted on 12 Jan 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Spatio-Temporal Deep Learning for day-ahead wind speed forecasting relying on WRF predictions

E. Christoforou^{1,2,3} · I.Z. Emiris^{2,1,3} ·
A. Florakis^{1,3} · D. Rizou⁴ · S. Zaharia⁴

Abstract We focus on deep learning algorithms, improving upon the Weather Research and Forecasting (WRF) model, and we show that the combination of these methods produces day-ahead wind speed predictions of high accuracy, with no need for previous-day measurements. We also show that previous-day data, if available, offer a significant enhancement. Our main contribution is the design and testing of original neural networks that capture both spatial and temporal characteristics of the wind, by combining convolutional (CNN) as well as recurrent (RNN) neural networks. The input predictions are obtained by a WRF model that we appropriately parameterize; we also specify a grid adapted to each park so as to capture its topography. Training uses historical data from 5 wind farms in Greece, and the 5-month testing period includes winter months, which exhibit the highest wind speed values. Our models improve WRF accuracy on average by 19.4%, and the improvement occurs in every month; expectedly, the improvement is lowest for the park where WRF performs best. Our neural network is competitive to state-of-the-art models, achieving an average MAE of 1.75 m/s. Accuracy improves for speed values up to 20 m/s, which are important in wind energy prediction. We also develop

¹Department of Informatics & Telecommunications, National & Kapodistrian University of Athens, Athens, Greece

²ATHENA Research Center, Maroussi, Greece

³Meteorologica P.C., Athens, Greece

⁴Renewable Energy Assessment & Forecasting Models, TERNA Energy, Athens, Greece

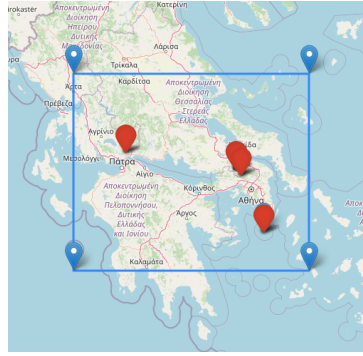


Fig. 1 Map of Greece: 5 case study parks (red) in a high-resolution WRF rectangle with corners (20.98, 39.38), (20.98, 36.99), (24.62, 39.38), (24.62, 36.99).

an RNN model and show that MAE reduces to less than 1 m/s for short-term predictions if actual data is employed.

Keywords Machine learning · neural network · Numerical Weather Prediction · long-term forecast · data modeling · wind speed · wind energy

1 Introduction

Wind is a renewable energy source of paramount importance for society thanks to its minimal environmental impact. Wind forecasting is crucial as it permits the power network to be prepared for variations in production, but also in optimizing energy trading [11]. An additional motivation includes site selection of wind farms, since the latter constitute an increasingly complex and costly investment [16]. In 2018, global wind power capacity grew by 9.6% to 591 GW and yearly wind energy production grew by 10%, reaching 4.8% of the world’s electric energy consumption [4], while providing 14% of the electricity in the European Union. The latter share rose to 15% in 2019 [22].

We concentrate on the Greek market for various reasons. First, it goes through a critical transitioning to the European target model which prescribes that Renewable Energy Source producers or aggregators are reimbursed through market-based mechanisms but are subject to balancing costs if production forecasts differ from actual output. Second, we report on an ongoing collaboration with industrial partners in Greece, namely Meteorologica P.C., and Terna Energy, the largest company in the sector, with presence in USA and Southeastern Europe. Terna provided actual data from 5 farms (Figure 1). The terrain complexity is a third reason for concentrating on Greece, since it is characterized by a mountainous profile and the interplay of land and sea.

We employ standard metrics for statistical evaluation: MAE (Mean Absolute Error), MAPE (Mean Absolute Percentage Error), WAPE (Weighted Absolute Percentage Error), MSE (Mean Squared Error), and RMSE (Root Mean Squared Error), which are defined below, where y represents the actual

and x the predicted values for a test set of cardinality n . It is known that $MAE \leq RMSE \leq \sqrt{n} MAE$.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - x_i|, \quad MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - x_i}{y_i} \right|, \quad WAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - x_i}{\frac{1}{n} \sum_{j=1}^n y_j} \right|, \quad (1)$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - x_i)^2, \quad RMSE = \sqrt{MSE}. \quad (2)$$

1.1 Previous Work

Let us review supervised learning methods that have shown strong performance in forecasting wind speed as well as energy [8].

Wind speed forecasting can be distinguished into very-short (seconds to 30 min), short (up to 6 hrs), medium (up to a day), long (up to a week), and very-long term forecasts. Most of the referenced methods use as input the timeseries of wind speed measurements, with AI models usually relying on recurrent neural networks (RNN) [1], such as LSTM or Elmann RNN [21], meant to capture time-dependent phenomena. One RNN for hourly forecasting in a 6-hr horizon with 12-hr input achieved a MAE of 1.18 m/s but used a testing period of only 45 days [7]. More standard architectures, like fully connected or convolutional neural networks (CNN) are also employed [5, 20].

Day-ahead forecasting is our focus. One LSTM model forecasts speed for a 24-hr horizon [19]. Using both actual data and Weather Research and Forecasting (WRF) predictions for training and forecasting, it achieves a MAE of 1.66 m/s. In the sequel, we combine it with our DSTNN to yield an ensemble that improves the accuracy of DSTNN.

A Multi Layer Perceptron (MLP) predicted speed for a 48-hr horizon using actual data, with MAE from 1.45 to 2.2 m/s [17]. Another MLP offered a 24-hr forecast, with RMSE between 0.997 and 1.803 m/s in Tilos, Greece [14]. A CNN used hourly measurements from 7 days to predict hourly the following 3 [10] achieving a MAE of 0.80 m/s. Speed values are mostly in the range of 1.5 to 4 m/s, which may explain the good accuracy.

When meteorological data is available, accuracy is increased. Using feed-forward neural networks with hourly wind speed and direction, and another 3 variables, very short term predictions achieved a MAE of 0.0212 [2]. This high accuracy concerns very low speeds, mostly 2 to 4 m/s. After training with meteorological data, an MLP using real data achieved an MSE of 5.27 to 7.05 m²/s² for a 24-hr forecast [6].

We develop methods that bypass the need of previous day measurements, which is the case for remote parks or when data fails to be transmitted on time: It has been a key characteristic in our collaboration with several large industrial producers. Meteorological predictions, of course, introduce their own error. An ensemble mixture density network achieved a MAE of 1.66 m/s in a

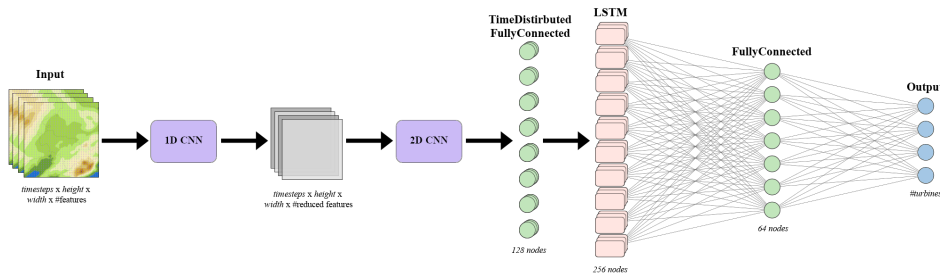


Fig. 2 The DSTNN architecture is based on a Time Distributed CNN followed by an LSTM.

72 hr horizon [13] using WRF predictions during 8 days: the first 5 for training, and only 3 for testing. A more relevant approach employs an optimized WRF model for training and forecasting [18], followed by a feed-forward network trained during 12 months in 2013, tested from February to December 2014. WRF and neural networks exhibit an MSE of 6.5 and 2.9 m^2/s^2 respectively, thus a 55% improvement.

1.2 Our Contributions

We develop supervised learning methods for wind speed forecasting in hourly time-steps, for a 24-hr horizon. Our main contribution is an original Deep Spatio-Temporal neural network (DSTNN) (Figure 2). Our implementation may also offer 48-hr forecasting, even in 15-min intervals. DSTNN forecasts speed at every turbine per park. Experiments reveal that even for turbines a few hundred meters apart, there can be a significant difference on the wind speed and, subsequently, on wind energy generation. Furthermore, in forecasting wind energy, individual turbines exhibit different availability.

DSTNN combines CNN and RNN, in order to capture, respectively, geospatial and temporal aspects of wind. The use of RNN was inspired by their success in capturing general time-series [3]. Our model avoids actual measurements but relies on WRF predictions provided by a two-nested scheme, with increased resolution in the rectangle of interest (Figure 1). Our WRF model has been appropriately parameterize and a grid has been adapted to each park.

We train our models with WRF forecasts as input, using historical data as targets from 5 farms of Terna, and test our approach over a period longer than 5 months including winter, which exhibits highest speed values. Our results (Table 2) significantly improve WRF predictions by up to 25%, and on average by 19.43%. It is a feature of our approach that the improvement occurs in every month, and it is lowest for the farm where WRF performs best (labeled SG). Our algorithms are competitive to state-of-the-art day-ahead forecasting, since they achieve an average MAE of 1.75 m/s, ranging from 1.53 to 2.1 m/s, and average RMSE 2.3, ranging from 2.02 to 2.65. Error is lower

for speed values up to 20 m/s that are most relevant to wind energy forecast. Our operational model provides Terna with daily wind speed predictions.

Another contribution is an LSTM neural network for short-term forecasting, relying on previous day measurements but no WRF; it mainly serves for juxtaposing it to the first scenario. It achieves average MAE of 0.94 m/s, ranging from 0.8 to 1.05 m/s. Lastly, the LSTM network of [19] has been combined with our DSTNN in an ensemble model, thus improving the forecasting accuracy (Section 4).

The next section presents how we obtain WRF predictions, and how we process them, as well as our deep neural networks in Subsection 2.3. Section 3 analyzes our results statistically; Subsection 3.1 revisits existing approaches, comparing their results to ours. Section 4 concludes with future work. Appendix A contains details on WRF and statistical results.

2 Data and Methods

2.1 WRF

Numerical Weather Prediction (NWP) is one of the most important tools for weather forecasting. NWP takes current observations of weather and processes them in order to forecast weather by Computational Fluid Dynamics (CFD). Using the corresponding equations, and parameterizations of other physical processes, CFD solvers integrate initial and boundary conditions and run over a geographic domain. Weather observations are used as input, and meteorological variables are predicted.

We leverage the Weather Research and Forecasting (WRF), a standard and widely used atmospheric model. We have parameterized our models and we are executing them on a rectangular region covering a large fraction of mainland Greece, including Central Greece and most of Peloponnese (marked by blue markers on Figure 1). Our model is two-nested since a finer-resolution model is embedded in a coarser-resolution parent model. The former is the aforementioned region in Greece, whereas the parent domain is the entire continent. Nesting allows running at finer resolution over the area of interest without requiring uniform resolution over a large domain. The main features of our implementation include a new Digital Elevation Model meant to account for better topography, while exploiting the two-way nesting capability. Certain physics schemes are introduced to model the lowest part of the atmosphere, coupled with a different Planetary Boundary layer scheme. Further details are part of the proprietary parts of the model.

Every WRF grid point captures meteorological patterns, such as temperature, which varies with altitude, or wind, which may be affected by the landscape. We obtain predictions at all points of an orthogonal grid whose resolution is 1 km. The wind farms are given and consist of a varying, yet known, number of wind turbines with given coordinates (longitude and latitude). Taking advantage of this knowledge, we downscale our original WRF

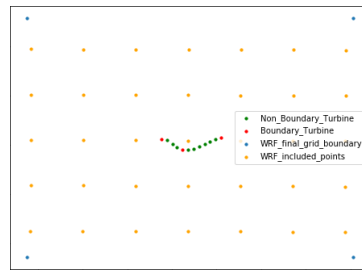


Fig. 3 Park with 3 extremal (red), and another 9 (green) turbines. The rectangle, defined by 4 corners (blue), includes 35 WRF points (orange). The axes show longitude and latitude.

grid, adapting it to the relevant area of the park, thus creating a targeted grid for the area of interest. WRF predicts 22 meteorological variables (Table 4): wind speed components in X-, Y-, and Z-axis, temperature and pressure, all available at 4 heights, and snow related features (snow water equivalent, daily total snow and ice).

Predictions are computed in a 1-hr frequency, based on station observations taken at 10-min intervals. Observation values below sensor thresholds are discarded; for instance, the wind speed threshold is at 0.1 m/s. Our model is currently being enhanced to output other parameters, such as meteorological variables at larger heights, so as to capture complex wind behaviour. In addition, we are planning to increase resolution by redefining the grid with a new edge length of 500 m as well as frequency down to 10-min intervals.

Let us present the construction of a rectangular grid per park, to include its turbines and an appropriate number of WRF points (Algorithm 1, Appendix): One procedure (FINDPARKBORDERS) specifies a minimum axis-aligned rectangle, namely *parkBox*, enclosing all park turbines. We define an outer rectangle, called *WRFbox*, to include *parkBox*, and parameterized by *STEP*, the margin in longitude and latitude around *parkBox*. The larger *STEP* is, the more comprehensive is the model but also the higher is the complexity; more importantly, if *WRFbox* is too large, it may include irrelevant geographical elements like the outer slope of a nearby hill. Another procedure (COLLECTWRFGRIDPOINTS) collects the WRF points forming *WRFbox*. In the park shown in Figure 3 there are 3 extremal out of 12 turbines, and 35 WRF grid points in *WRFbox*. Each of its edges is defined by moving away from the corresponding edge of *parkBox* by $STEP = 0.04$ degrees. For the other parks, we experimentally set $STEP = 0.04$ or $STEP = 0.035$ hence including 2 or 3 rows and columns of WRF grid points beyond *parkBox*.

2.2 Data Modeling and Preprocessing

Training uses actual wind speed as target, while the LSTM also uses them as input. The provider is Terna: speed data were retrieved from a Supervisory Control and Data Acquisition (SCADA) system of wind turbines, used for

Table 1 Wind farms of Terna Energy, used for training and testing our models.

Park Name	Code	Training Data Period	Tower heights (m)
Saint Georgios	SG	9/10/2016 - 1/9/2019	80, 84
Krekeza	K	31/12/2015 - 1/9/2019	80
Louzes	L	31/12/2015 - 1/9/2019	67
Rachoula II	RA-2	31/12/2015 - 1/9/2019	80
Rachoula III	RA-3	28/2/2018 - 1/9/2019	80

remote supervision and control of the turbines. The wind speed recordings are 10-min averages of 1-Hz sampled values, provided for each wind turbine of each farm; farms are listed in Table 1 and contain from 3 or 4, up to 12 or 23 turbines. The second class of data consists of meteorological predictions. The number of WRF variables is 22, so this type of input consists of a [grid_height x grid_width x 22] table, where the first two specify the size of *WRFbox*. All variables are input to DSTNN, since no feature importance/selection algorithm has been employed yet.

Let us examine data preprocessing. Since wind speed timeseries hide strong temporal patterns, we pack 6 consecutive WRF grids, thus creating timeseries of grids. Our training target is the value of the 6th hour. Therefore, WRF data have the shape of [6 x grid_height x grid_width x 22] and targets are vectors of the form [1 x #turbines]. Actual wind is provided in 10-min resolution and gets aligned with the respective 1 hr WRF resolution, since we produce hourly predictions. Thus, we resample the data and compute each hour-measurement either as the mean of the 3 previous and the 3 following timestamps, or the mean of the 6 previous ones.

The dataset suffers from missing values (4-8% on average). In some cases, the missing values cover consecutive days or even weeks, so their handling is hard. Two approaches have been adopted. The simplest one was to exclude all records that contain missing values for some of the turbines. This reduces the size of the training data. The second approach is to interpolate missing values. Linear interpolation is implemented by means of curve-fitting, using linear polynomials to construct new data points within the range of a discrete pointset. The method gives good results for small gaps and, at the same time, preserves the sequential data behavior. However it fills large (e.g. month-long) gaps inaccurately. Both approaches give promising results, but more sophisticated gap filling may have to be employed in the future. For scaling, both actual wind and WRF values are normalized per variable, within [0, 1], according to the Min-Max formula: given x , $m \leq x \leq M$, its scaled value is $(x - m)/(M - m)$.

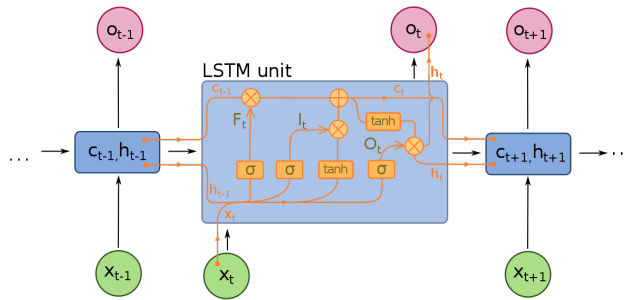


Fig. 4 Unfolding of an LSTM (from wikimedia.org).

2.3 Neural networks

In this section, we describe our novel Deep Spatio-Temporal neural network (DSTNN) for long-term wind speed forecasting, based on WRF data. Our second contribution concerns a short-term model using timeseries of wind speed data and no meteorological predictions, based on an RNN.

The DSTNN architecture (Figure 2) consists of two parts: a CNN and an RNN sub-network for capturing, respectively, both spatial and temporal patterns. In addition, there are more standard layers, namely fully connected (dense) layers. Since the input is initially given as timeseries, we aim at architectures that extract temporal information taking advantage of the underlying time-related wind patterns: RNNs have been designed for such tasks, e.g. in natural language processing, with significant success. Their architecture handles, classifies, or forecasts properties of sequences; in addition, they are capable to produce sequences with certain desired properties.

Among numerous existing RNNs, the LSTM [9] model is used since it addresses the vanishing gradient problems of traditional RNNs, when sequences get long. LSTMs not only keep adjacent temporal information, but also control long-term information by means of a controlling gate. A common LSTM unit is composed of a cell, and an input, an output and a forget-gate (Figure 4). The cell remembers values over arbitrary time intervals and the 3 gates regulate information flow in and out of the cell. Initially, the forget-gate extracts the information to be preserved from the prior state. Then, the input gate determines how much “current” information should be used as input in generating the current state. The output gate filters the information deemed significant. This is repeated in every time step of sequential processing, allowing LSTM memory to remember or forget cell states.

Furthermore, every step of the series contains spatial information, because the WRF forecasts are available for every geographical point on a grid. In this way, each step of the series is represented by a grid that contains the respective park. Every grid is processed by a CNN, thus capturing neighborhood and spatial patterns of the covered area. The output is a vector of size $[1 \times \#\text{turbines}]$, where the latter denotes the number of turbines in the park.

Specifically, DSTNN starts with a Time Distributed (TD) CNN in order to overcome modelling uncertainty issues, such as wind speed extrapolations, which are usually the main contributor of uncertainty in complex terrains [15]. Each step of the input series consists of a grid of WRF points, so the input has shape [timesteps x grid_height x grid_width x 22]. The TD structure implies that an individual CNN is applied to every temporal slice of the input.

Every grid-step is processed by two separate CNN layers: a 1d CNN performing convolution in feature space, and a shallow 2d CNN extracting neighborhood patterns from the WRF grid. The first applies dimensionality reduction to the input features. The second has 32 convolutional 3 x 3 kernels followed by a 2 x 2 max pooling layer. The small pooling window and small convolutional filters avoid information loss. Both convolutional kernels are initialized from the Gaussian distribution, with 0 mean and standard deviation 0.08. After each CNN, we apply a ReLU (Rectifier Linear Unit), a standard activation function.

A TD dense layer with 128 nodes follows, in order to reduce the dimension of the features, and the output is of the form [timeseries_length x num_CNN_extracted_features]. This is input to an LSTM [9] with 256 nodes, with recurrent dropout rate of 0.15 (dropping probability). A final dense layer with 64 nodes performs the final regression. The implemented architecture has the advantage of processing two differently formatted inputs consequently and in a different manner, thus extracting hidden patterns from both of them.

The RMSprop optimizer is employed since it is known to show good performance with RNNs [7]. The learning rate equals 0.001, which is a standard value during optimization and was found, by hyperparameter search, to be appropriate. The loss function is MSE, and the batch size is 64, chosen through experimentation. Training was undertaken using an Early Stopping Criterion, so it was terminated after 20 epochs of no improvement of the validation loss, compared to the best loss that had already been achieved. Hyperparameter selection employed grid search.

Stateless RNN has been used in most experiments; the hidden and the cell states of the LSTM are re-initialized for every batch. This was preferred over the statefull approach because our inputs are not provided in a perfect chronological order, due to the large number of missing values. Concerning the use of available historical data, 70% has been chosen for the training of the model. Fractions of 15% each have been considered as validation and test sets. The length of the timeseries is 6-12 steps, and the nodes of every layer are on average 64-128. We have developed implementations on Tensorflow, which is lower-level, and on Keras, which is more user friendly, using Tensorflow's back-end. The model deployed for daily operation is the latter.

Our second contribution is a short-term neural network developed for juxtaposing with DSTNN, offering hour-ahead predictions. It employs solely actual wind-speed timeseries per turbine, both for training and forecasting, but no meteorological data. This model was trained with timeseries of 6 hourly steps namely 6 measurements in the previous 6 hours. The input has overall shape of [timesteps x #turbines], and the output is a vector of shape [1 x

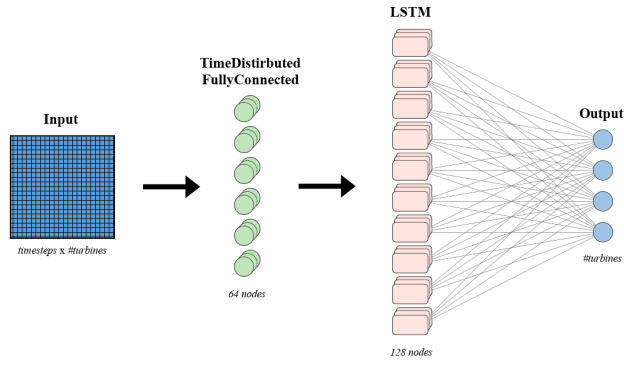


Fig. 5 LSTM model including 3 layers: TD fully connected, LSTM, and fully connected.

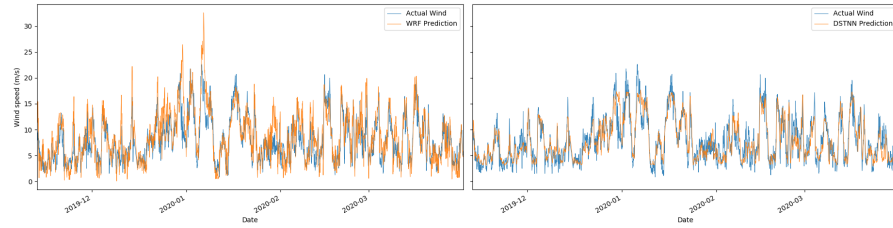


Fig. 6 Forecasting (orange) of WRF (left) and DSTNN (right) against actual measurements (blue) for Krekeza. The axes show date (x) and wind speed (y).

#turbines]. We predict the next hour, for every turbine of the park. This is standard regression. Our model is based on LSTM, and is comprised of 3 parts (Figure 5): (1) a TD fully connected layer of 64 nodes, applied to each temporal slice of the initial series, (2) an LSTM layer with 128 nodes and recurrent dropout rate equal to 0.2, and (3) a final fully connected layer, whose number of nodes equals the number of turbines of the park, and which performs the final regression.

3 Case study and Discussion

We examine results on 5 wind parks in Greece representing very different terrains, since they include an island (Saint Georgios), mountainous locations (Louzes, but also Krekeza), vicinity to a coastal environment (Krekeza), as well as inland hilly locations (Rachoula). They lie at different elevations, from sea level (SG) to almost 700 m altitude (Rachoula). Louzes is spread over two neighbouring slopes, making the park's topography quite complex, which is probably the main reason why it has eluded accurate modeling until now. The coordinates of the WRF grids per park are in Table 5 (Appendix).

Figure 6 shows graphical results of DSTNN's forecasting against the WRF's for Krekeza, one of the hardest farm to model. The improvement of DSTNN

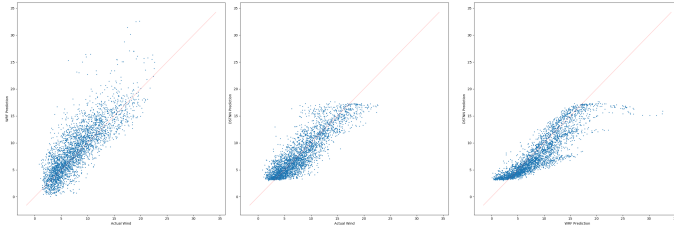


Fig. 7 Actual speed (x-axis) against WRF (left) and DSTNN predictions (middle), and WRF (x-axis) against DSTNN predictions (right). In the first two, distance of a point from the (red) diagonal is inversely proportional to accuracy. At right, it is inversely proportional to similarity of the two predictions.

over WRF predictions is obvious. As it is usually observed in statistical analysis of forecasting, e.g. [12], classic models such as WRF may fail to capture extreme values of meteorological variables. These are precisely the cases that DSTNN manages to capture more accurately, hence offering a clear advantage over WRF. Figure 10 (Appendix) offers the same comparison for the remaining parks, leading to the same conclusion.

Figure 7 shows clearly that DSTNN avoids outputting very low (< 2.9 m/s) and very high (> 17.7 m/s) speed values, thus improving accuracy. This is also validated by Table 3, where DSTNN improves MAE for the lower and upper speed range buckets in Krekeza, especially for 0-5 m/s and 15-20 m/s. Specifically for the upper speed buckets, WRF appears to mostly overpredict wind speed, increasing MAE significantly. DSTNN seems to apply upper and lower thresholds to the WRF predictions, while also improving predictions for intermediate speeds. This behavior of DSTNN is also seen in Figure 7 (right), where higher and lower speed predictions by WRF are limited by DSTNN, while predictions for intermediate speed values are tend to be almost equal.

Table 2 Performance of WRF, LSTM with actual data, DSTNN and improvement of DSTNN over WRF; all testing periods end on 31/3/2020

Park	Period start	WRF (long-term)			LSTM (short-term)			DSTNN (long-term)			reduce MAE%
		MAE	WAPE	MSE	MAE	WAPE	MSE	MAE	WAPE	MSE	
SG	24/10/19	1.99	25.1	7.1	1.05	13.5	1.95	1.77	22.4	5.4	11.1
K	13/11/19	2.20	25.0	8.4	0.95	11.05	1.60	1.77	20.5	5.6	20.0
L	16/11/19	2.54	34.0	11.2	0.99	14.2	1.64	2.10	27.7	7.0	17.3
RA2	16/12/19	2.10	28.0	8.0	0.92	13.4	1.65	1.60	21.2	4.5	23.8
RA3	16/12/19	2.03	28.3	7.56	0.80	12.3	1.40	1.53	21.3	4.1	25.0
mean	-	2.17	28.1	8.5	0.94	12.89	1.65	1.75	22.6	5.3	19.4

Results in Table 2 quantify the performance of our models daily for a testing period of more than 5 months, although for the smallest parks (RA-2, RA-3) this period reduces to 3.5 months. They are winter months and March, which are the windiest months in the northern hemisphere hence the most interesting for wind energy production; for the 3 parks with longer testing periods, the latter include at least the second half of November whereas, for

Table 3 Statistics per wind speed range

	park	0-5 m/s	5-10 m/s	10-15 m/s	15-20 m/s	>20 m/s	Total
#timestamps	SG	1203	1430	837	306	40	3816
WRF MAE		1.56	1.79	1.65	1.83	2.94	
DSTNN MAE		1.47	1.57	1.79	1.64	3.87	
#timestamps	K	897	1298	752	341	48	3336
WRF MAE		1.63	1.87	2.03	2.9	3.4	
DSTNN MAE		1.23	1.56	2.14	1.87	3.04	
#timestamps	L	978	1372	622	131	13	3116
WRF MAE		1.9	2.2	2.9	3.2	4.1	
DSTNN MAE		2.1	1.5	2.2	3.6	5.4	
#timestamps	RA-2	801	1149	493	102	23	2568
WRF MAE		1.7	2.0	2.2	2.6	3.5	
DSTNN MAE		1.2	1.4	1.8	2.6	4.5	
#timestamps	RA-3	788	1252	451	73	4	2568
WRF MAE		1.64	2.0	2.1	2.6	1.0	
DSTNN MAE		1.42	1.25	1.7	2.6	6.3	
#timestamps	Total	4667	6501	3155	953	128	

SG, the testing periods starts on October 24th. WRF error is calculated as the difference of actual wind value measured at each turbine and the wind prediction at the nearest WRF grid point, using Euclidean distance. We report on MAE, WAPE, and MSE, calculated for each turbine of every park.

For a comprehensive comparison with the state-of-the-art, which typically exploits actual measurements, we also experiment with our LSTM-based next-hour model. It achieves average MAE of 0.94 m/s, ranging from 0.8 to 1.05 m/s, using the same 5 parks, same historical data and same testing periods as for DSTNN (Table 2). Clearly, previous-day measurements can drastically enhance the accuracy of DSTNN. Of course, improved accuracy is achieved on a much easier task than for WRF and DSTNN.

DSTNN improves upon WRF on all parks, under all metrics; see the table’s last column. The improvement is smaller on the “easiest” park (island SG), where WRF performs best, and on the “hardest” one (L), where the terrain’s complexity makes both models perform worst. DSTNN and LSTM are competitive to the state-of-the-art; this is especially true for DSTNN against long-term models that do not rely on actual wind measurements.

A more detailed analysis (Figure 8) explores the evolution of MAE, roughly on a monthly basis. The intervals are given in Table 6 (second column, Appendix) and they are 3 to 5, depending on the park. All first periods are somewhat longer than one month. By comparing the average MAE of our DSTNN against WRF for different periods (months), it is verified that the former improves WRF’s forecasting on all time periods, and the improvement does not seem to depend on the month. Another important observation is that MAE seems to be maximum in January, whereas in SG it is maximized in December and March; however, given statistics of a single year only, it is hard to draw general conclusions. A comparative performance using MAE, WAPE, and MSE on the same intervals is in Table 6 (Appendix).

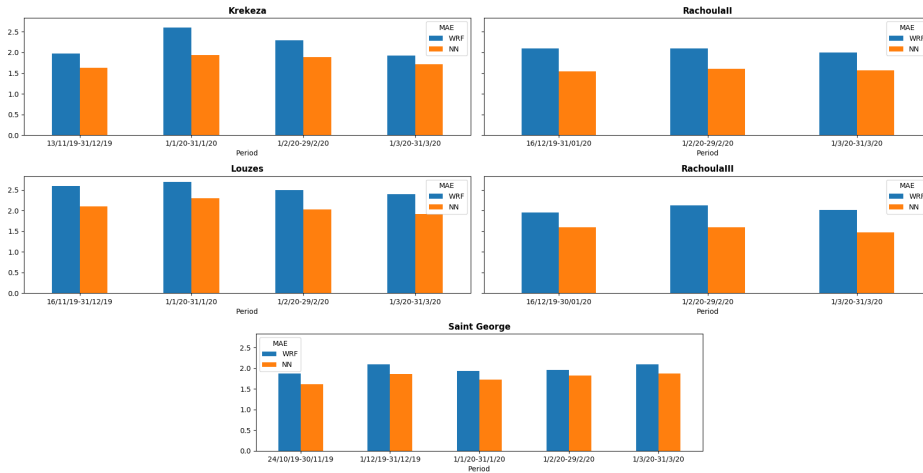


Fig. 8 Periodical evolution of MAE for the WRF model (blue) against DSTNN (orange) for each park.

Figure 9 illustrates the MAE of DSTNN against WRF for a partition of all wind speed values over a set of 5 non-overlapping, wind ranges, namely: 0-5, 5-10, 10-15, 15-20, >20 m/s. These statistics are significant for the wind energy market, mostly interested in speeds ranging from 5 to 20, or even 25 m/s. The total number of hourly timestamps ranges from 2568 (RA-2, RA-3) to 3816 (SG), as reported in Table 3. Here, the DSTNN improves the WRF predictions in almost all cases. The exceptions are cases where wind speed is higher than 20 m/s for all parks besides K. This is explained by the very small number of instances with such wind speeds, namely up to 40. Also outlier exceptions for low (L), mid-range (K, SG) and high (L, RA-2, RA-3) wind speeds.

Table 3 offers further statistics, such as number of timestamps per wind speed range, over all parks. Expectedly, MAE for both WRF and DSTNN roughly grows with wind speed, since larger values exhibit larger absolute error. An important observation is that MAE for WRF as well as DSTNN is maximized for speed > 20 m/s, which is expected. Hence it is lower than average for values ≤ 20 m/s which are very relevant to wind energy prediction.

3.1 Discussion

We juxtapose our results to results in the most relevant papers discussed in Section 1.1, although test data is different in every case.

First, it is very encouraging that DSTNN has comparable accuracy to long-term prediction methods using previous-day meteorological measurements: For 24-hr [6, 14] and 48-hr predictions [17], MLP-based neural networks report an MSE of 5.27 to 7.05 m^2/s^2 , an RMSE between 0.997 and 1.803 m/s, and a MAE between 1.45 and 2.2 m/s, respectively. These are comparable to

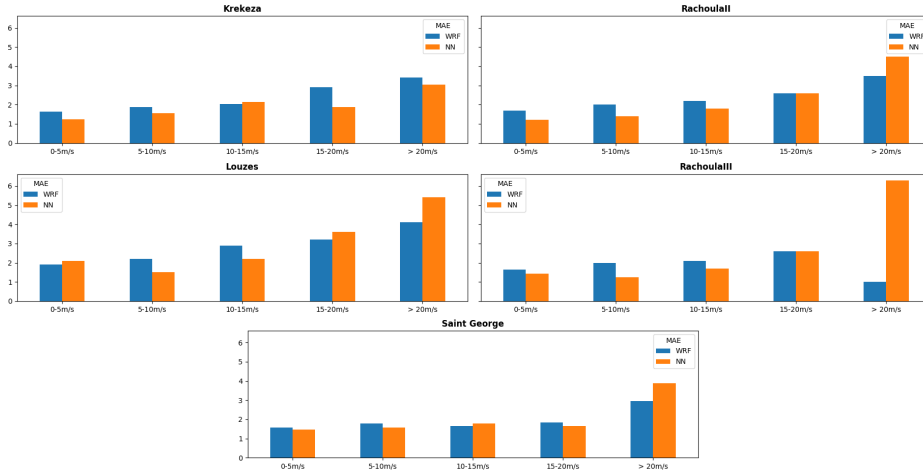


Fig. 9 MAE for 5 wind ranges of WRF (blue) against DSTNN (orange) per park.

DSTNN’s mean MSE of $5.3 \text{ m}^2/\text{s}^2$, RMSE of 2.3 m/s and MAE of 1.75 m/s . We conclude that the predictive power of DSTNN is superior since accuracy is comparable but without any actual measurement during forecasting. This motivates us to explore deeper neural networks and advanced data modeling.

Only a limited number of models rely exclusively on WRF. For 72-hr predictions, a MAE of 1.66 m/s [13] is not significantly better than DSTNN and has been tested during 3 days only. A relevant approach uses WRF forecasts post-processed by a neural network [18] achieving, respectively, an MSE of 6.5 and of $2.9 \text{ m}^2/\text{s}^2$. Their long testing period of 11 months partly explains the quality of their results. Our WRF and DSTNN yield a mean MSE of 8.5 and $5.3 \text{ m}^2/\text{s}^2$ respectively, but our experiments concern winter months where absolute error peaks. We may conclude that our main weakness lies in WRF, whereas DSTNN seems quite accurate. We currently concentrate on a finer resolution and smaller time-step of WRF.

Ensemble learning is based on combining multiple AI models, in view of exploiting the best one in different time periods. We tested an ensemble of DSTNN with the simple LSTM model [19] discussed at Previous Work for day-ahead forecasting on the aforementioned parks. They process independently WRF predictions, while the second network also uses actual wind. A final fully connected layer combines the two outputs. The ensemble improved each individual model for wind speeds in the range of 10 to 15 m/s, by 5%. In other ranges, the improvement is smaller. In the immediate future we shall report on advances with ensemble models.

Lastly, our LSTM model for short-term prediction using 6 hourly measurements is more accurate than an RNN forecasting 6 hours given 12 [7]. This is another promising avenue.

4 Conclusion

A Deep Spatio-Temporal neural network for wind speed forecasting relying exclusively on the WRF model was presented. Results on 5 wind parks in Greece demonstrate a significant improvement of day-ahead WRF on average by 19.43%, while being competitive to the state-of-the-art models, having an accuracy with an average MAE of 1.75 m/s. Forecasting by just previous-day data with an LSTM achieves an average MAE of 0.94 m/s.

A challenging task is to transfer pre-trained neural networks to parks that lack a sufficient amount of historic data, a common real life situation in production settings. By retraining using any available data, the results seem promising, and the models appear to generalize on the patterns of the wind. Data science methods may be used to improve data cleaning, outlier handling and gap filling. Current work to adapt our models to energy prediction achieve normalized MAPE of 14% on a single farm, for a 6-month testing period. Lastly, our grid-based approach can be transferred to photovoltaic energy prediction.

Acknowledgements Research partially funded by project “PeGASUS: Approximate geometric algorithms and clustering with applications in finance” (MIS 5047662) under call “Support for researchers with emphasis on young researchers: cycle B” (EDBM103), co-financed by Greece and the EU (European Social Fund) under the operational program Human Resources Development, Education & Lifelong Learning 2014-20. We thank Meteorologica P.C. for offering hardware infrastructure and guidance on meteorological aspects.

References

1. Balluff, S., Bendfeld, J., Krauter, S.: Short term wind and energy prediction for offshore wind farms using neural networks. In: Intern. Conf. Renewable Energy Research & Applications, pp. 379–382 (2015)
2. Barhmi, S., Elfatni, O., Belhaj, I.: Forecasting of wind speed using multiple linear regression and artificial neural networks. *Energy Systems* **11** (2020). DOI 10.1007/s12667-019-00338-y
3. Christoforou, E., Emiris, I.Z., Florakis, A.: Neural networks for cryptocurrency evaluation and price fluctuation forecasting. In: P. Pardalos, I. Kotsireas, Y. Guo, W. Knottenbelt (eds.) *Mathematical Research for Blockchain Economy*, pp. 133–149. Springer International Publishing, Cham (2020)
4. Council, G.W.E.: 51.3 GW of global wind capacity installed in 2018. <https://gwec.net/51-3-gw-of-global-wind-capacity-installed-in-2018/> (2019)
5. Díaz, D., Torres, A., Dorransoro, J.R.: Deep neural networks for wind energy prediction. *Advances in Computational Intelligence* pp. 430–443 (2015)
6. Finamore, A.R., Calderaro, V., Galdi, V., Piccolo, A., Conio, G.: A wind speed forecasting model based on artificial neural network and meteorological data. In: Proc. IEEE 16th Intern. Conf. Environment and Electrical Engineering, pp. 1–5 (2016)
7. Ghaderi, A., Sanandaji, B.M., Ghaderi, F.: Deep forecast: Deep learning-based spatio-temporal forecasting. *ArXiv abs/1707.08110* (2017)
8. Godinho, M., Castro, R.: Comparative performance of AI methods for wind power forecast in Portugal. *Wind Energy* (2020). URL <https://onlinelibrary.wiley.com/doi/pdf/10.1002/we.2556>
9. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural computation* **9**, 1735–80 (1997). DOI 10.1162/neco.1997.9.8.1735

10. Huang, C.J., Kuo, P.H.: A short-term wind speed forecasting model by using artificial neural networks with stochastic optimization for renewable energy systems. *Energies* **11**, 2777 (2018)
11. Huang, J., Lu, X., McElroy, M.: Meteorologically defined limits to reduction in the variability of outputs from a coupled wind farm system in the central us. *Renewable Energy* **62** (2014)
12. Mauch, B., Apt, J., Carvalho, P., Small, M.: An effective method for modeling wind power forecast uncertainty. *Energy Systems* **4** (2013). DOI 10.1007/s12667-013-0083-3
13. Men, Z., Yee, E., Lien, F.S., Wen, D., Chen, Y.: Short-term wind speed and power forecasting using an ensemble of mixture density neural networks. *Renewable Energy* **87**, 203–211 (2016)
14. Moustris, K.P., Zafirakis, D., Alamo, D.H., Nebot Medina, R.J., Kaldellis, J.K.: 24-h ahead wind speed prediction for the optimum operation of hybrid power stations with the use of artificial neural networks. *Perspectives on Atmospheric Sciences* pp. 409–414 (2017)
15. Papoutsakis, A., Zacharia, S., Kouvara, F., Karagiannis, D., Katsoulis, G., Daikos, A.: Wind model uncertainty analysis and validation over operating wind farms in complex terrain. *European Wind Energy Conference and Exhibition (EWEC)* **1**, 384–393 (2013)
16. dos Reis, M.M.L., Mazetto, B.M., da Silva, E.C.M.: Economic analysis for implantation of an offshore wind farm in the Brazilian coast. *Sustainable Energy Technologies and Assessments* **43** (2021)
17. Salcedo-Sanz, S., Pérez-Bellido, Á.M., Ortíz-García, E., Portilla-Figueras, A., Prieto, L., Paredes, D.: Hybridizing the fifth generation mesoscale model with artificial neural networks for short-term wind speed prediction. *Renewable Energy* **34**, 1451–1457 (2009)
18. Salfate, I., Marin, J.C., Cuevas, O., Montecinos, S.: Improving wind speed forecasts from the weather research and forecasting model at a wind farm in the semiarid coquimbo region in central chile. *Wind Energy* **23**(10), 1939–1954 (2020)
19. Tzamos, C.: Ventusnet: Deep learning for wind speed prediction. MSc Thesis, Dept Informatics & Telecoms, National and Kapodistrian University of Athens (Greece). <https://pergamos.lib.uoa.gr/uoal/oa/dl/object/2878206> (2019)
20. Wang, H., Li, G., Wang, G., Peng, J., Jiang, H., Liu, Y.: Deep learning based ensemble approach for probabilistic wind power forecasting. *Applied Energy* **188**, 56–70 (2017). URL www.sciencedirect.com/science/article/pii/S0306261916317421
21. Wang, J., Shanshan, Q., Zhou, Q., Jiang, H.: Medium-term wind speeds forecasting utilizing hybrid models for three different sites in xinjiang, china. *Renewable Energy* **76** (2014)
22. WindEurope.org: Wind energy in europe in 2018. <https://windeurope.org/about-wind/statistics/european/wind-energy-in-europe-in-2018> (2019)

A WRF information, and Statistical results

This section provides the meteorological variables predicted by WRF, the coordinates of the WRF rectangle per park, and pseudocode for computing these rectangles. Further statistical results of our predictions are given, as well.

Table 4 Meteorological variables predicted by WRF

Variable Name	Description	Units
$U_{60}, U_{80}, U_{100}, U_{120}$	X component of wind at 60, 80, 100, 120 m	m/s
$V_{60}, V_{80}, V_{100}, V_{120}$	Y component of wind at 60, 80, 100, 120 m	m/s
$W_{60}, W_{80}, W_{100}, W_{120}$	Z component of wind at 60, 80, 100, 120 m	m/s
$T_{60}, T_{80}, T_{100}, T_{120}$	Temperature at 60, 80, 100, 120 m	K
$PR_{60}, PR_{80}, PR_{100}, PR_{120}$	Pressure at 60, 80, 100, 120 m	hPa
$Snow$	Snow water equivalent	$\text{kg}\cdot\text{m}^{-2}$
$SnowNC$	Daily total snow and ice	mm

Table 5 Corner coordinates (longitude, latitude) of WRF rectangle per park

Park name	NW	SW	NE	SE
SG	(23.83, 37.53)	(23.83, 37.42)	(24.00, 37.53)	(24.00, 37.42)
K	(23.51, 38.21)	(23.51, 38.11)	(23.60, 38.21)	(23.66, 38.11)
L	(21.71, 38.49)	(21.71, 38.38)	(21.87, 38.49)	(21.87, 38.38)
RA-2	(23.42, 38.28)	(23.42, 38.19)	(23.56, 38.28)	(23.56, 38.19)
RA-3	(23.44, 38.28)	(23.44, 38.18)	(23.56, 38.20)	(23.56, 38.18)

Algorithm 1 Adaptive cropping of original WRF grid

```

1: Input:  $parkCoords[i], i = 1..m$  ▷  $m$  turbines' coordinates
2: Input:  $WRFp[i], i = 1..n$  ▷  $n$  WRF-points' coordinates
3:
4: procedure FINDPARKBORDERS ( $parkCoords$ ) ▷ min enclosing rectangle
5:    $N \leftarrow \max parkCoords[i].y; S \leftarrow \min parkCoords[i].y$ 
6:    $E \leftarrow \max parkCoords[i].x; W \leftarrow \min parkCoords[i].x$ 
7:    $park \leftarrow [N, S, E, W]$ 
8:   return  $park$  ▷ use turbines with max/min longitude/latitude
9: end procedure
10: procedure COLLECTWRFGRIDPOINTS ( $park, WRFp, STEP$ )
11:   ▷ WRF points at (longitude/latitude) distance STEP
12:    $WRFbox \leftarrow \emptyset$ 
13:   for every  $i = 1..n$  do
14:     if  $WRFp[i].y > parkBox.S - STEP$  &  $WRFp[i].y < parkBox.N + STEP$  &  $WRFp[i].x > parkBox.W - STEP$  &  $WRFp[i].x < parkBox.E + STEP$  then
15:        $WRFbox.add(WRFp[i])$ 
16:     end if
17:   end for
18:   return  $WRFbox$ 
19: end procedure

```

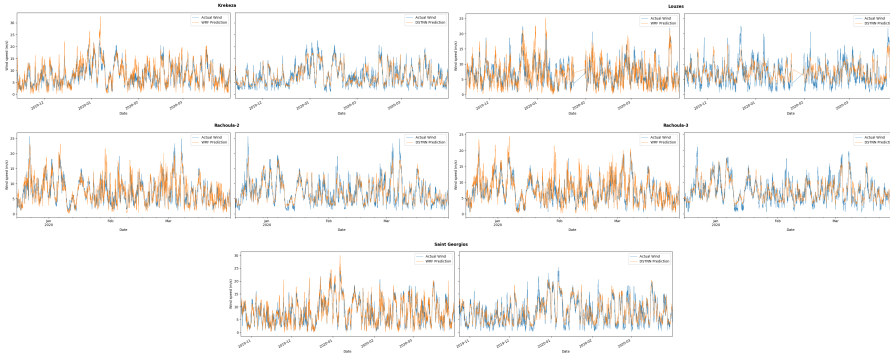


Fig. 10 Forecasting (orange) of WRF (left) and DSTNN (right) against actual values (blue). DSTNN improves WRF in all parks: Krekeza (top-left), Louzes (top-right), Rachoula-2 (middle-left), Rachoula-3 (middle-right) and Saint Georgios (bottom).

Table 6 Statistics per month-long intervals

Park	Period	WRF			DSTNN		
		MAE	WAPE	MSE	MAE	WAPE	MSE
SG	24/10/19-30/11/19	1.87	29.50	6.0	1.61	25.10	4.5
	1/12/19-31/12/19	2.1	24.50	7.9	1.86	22.10	6
	1/1/20-31/1/20	1.94	20.20	7.0	1.73	17.60	5
	1/2/20-29/2/20	1.96	25.00	7.15	1.82	23.00	6
	1/3/20-31/3/20	2.1	27.50	8.0	1.87	24.20	6
K	13/11/19-31/12/19	1.97	26.00	7.4	1.63	21.20	4.9
	1/1/20-31/1/20	2.6	24.60	11.1	1.94	19.00	6.5
	1/2/20-29/2/20	2.3	25.50	8.7	1.89	21.20	5.8
	1/3/20-31/3/20	1.93	23.50	6.6	1.72	21.00	5.6
L	16/11/19-31/12/19	2.6	33.60	11.3	2.1	27.00	7.5
	1/1/20-31/1/20	2.7	38.00	14.5	2.3	33.00	8
	1/2/20-29/2/20	2.5	33.00	10.1	2.03	27.00	6.7
	1/3/20-31/3/20	2.4	33.00	9.8	1.92	27.00	5.8
RA-2	16/12/19-31/01/20	2.1	26.00	8.2	1.54	19.30	4.5
	1/2/20-29/2/20	2.1	29.40	7.7	1.6	23.10	4.7
	1/3/20-31/3/20	2	29.00	7.8	1.57	23.00	4.5
RA-3	16/12/19-30/01/20	1.96	30.00	7.3	1.6	24.00	4.5
	1/2/20-29/2/20	2.13	30.10	7.8	1.6	22.80	4.1
	1/3/20-31/3/20	2.02	26.10	7.6	1.47	19.10	3.8