Received: 23 September 2018 | Revised: 6 February 2019 | Accepted: 15 February 2019
DOI: 10.1111/exsy.12394

WILEY Expert Systems

SPECIAL ISSUE PAPER

Big data solar power forecasting based on deep learning and multiple data sources

José F. Torres[1] | Alicia Troncoso[1] | Irena Koprinska[2] | Zheng Wang[2] | Francisco Martínez-Álvarez[1]

[1]Data Science and Big Data Lab, Universidad Pablo de Olavide, ES-41013 Seville, Spain
[2]School of Computer Science, University of Sydney, Sydney, Australia

Correspondence
Alicia Troncoso, Data Science and Big Data Lab, Universidad Pablo de Olavide, ES-41013 Seville, Spain.
Email: atrolor@upo.es

Abstract

In this paper, we consider the task of predicting the electricity power generated by photovoltaic solar systems for the next day at half-hourly intervals. We introduce DL, a deep learning approach based on feed-forward neural networks for big data time series, which decomposes the forecasting problem into several sub-problems. We conduct a comprehensive evaluation using 2 years of Australian solar data, evaluating accuracy and training time, and comparing the performance of DL with two other advanced methods based on neural networks and pattern sequence similarity. We investigate the use of multiple data sources (solar power and weather data for the previous days, and weather forecast for the next day) and also study the effect of different historical window sizes. The results show that DL produces competitive accuracy results and scales well, and is thus a highly suitable method for big data environments.

KEYWORDS
big data, deep learning, solar power, time series forecasting

1 | INTRODUCTION

Solar energy is a very promising renewable electricity source that is still not fully utilized. Recently, there has been a rapid growth in the installed large-scale and residential (rooftop) solar photovoltaic (PV) systems. This is due to the reduced cost of solar PV panels, improvements in technology and performance, and government initiatives encouraging the use of solar systems.

As a result, in many countries now, the cost of electricity produced by solar energy is comparable with that of conventional energy sources. This competitive cost, coupled with the fact that solar is a clean and abundant energy source, has led to a huge growth in the generated solar energy. This trend is expected to continue—for example, by 2020, the global solar capacity is projected to reach 700 GW, an increase of about 140 times compared with 2005 (SolarPowerEurope, 2016). In Australia, it is expected that by 2050, 30% of the electricity supply will come from solar energy (Flannery & Sahajwalla, 2013).

However, solar energy is highly variable since it depends on meteorological conditions such as solar radiation, cloud cover, rainfall, and temperature. This dependency creates uncertainty about the amount of solar power that will be generated, which makes the integration of solar power into the electricity grid and electricity markets more difficult. Hence, the ability to accurately predict the generated solar power is a task of utmost importance and relevance for both energy managers and electricity traders, in order to minimize uncertainty and ensure reliable electricity supply at acceptable cost.

Historical PV solar power data with high frequency is easily available, and therefore, advanced computing technologies and machine learning approaches for big data can be used to analyse very large time series. Deep learning is an emerging branch of machine learning that extends the traditional neural networks by using architectures with many hidden layers that are able to learn hierarchical feature representations.

One of the main drawbacks of the classical neural networks is that if they have many hidden layers they become difficult to train (Livingstone, Manallack, & Tetko, 1997; Schmidhuber, 2015)

Deep learning involves the use of more effective learning algorithms and techniques to train neural networks with many hidden layers.

In this paper, we propose a new approach based on deep learning feed-forward neural networks to forecast short-term (one day ahead), big solar power time series data. Day ahead predictions are one of the most common industry-requested operational forecasts (Kostylev & Pavlovski,

https://doi.org/10.1111/exsy.12394

2011). They are needed for operational planning, switching sources, programming backups, short-term power purchases, and for planning of reserve usage and peak load matching (Ervural & Ervural, 2018; Reikard, 2009). Specifically, we consider the following task: given a time series of PV power outputs up to day *d*, where one day is a vector of half-hourly power outputs, our goal is to forecast the half-hourly PV power output for the next day $d + 1$.

We first compare the performance of our proposed DL algorithm with two other advanced methods for forecasting presented in (Wang, Koprinska, & Rana, 2017). In particular, we compare DL with the (a) Pattern Sequence-based Forecasting (PSF) algorithm, which uses clustering and similarity of patterns (Martínez-Álvarez, Troncoso, Riquelme, & Aguilar, 2011), and (b) a neural network-based model with one hidden layer (we will refer to it as NN), used as a reference method for solar power forecasting. Next, we conduct a scalability study in order to evaluate the suitability of all methods to deal with big data time series. We also analyse if the accuracy of DL and the methods used for comparison improves when using weather and weather forecast data as an additional input, taking into account different scenarios corresponding to different percentages of noise in the weather forecast data (10%, 20%, and 30%). Finally, we study how the size of the historical window affects the behaviour of our DL prediction system.

In summary, the main contributions of this work are:

1. We propose DL, a deep learning approach based on feed-forward neural networks, for predicting the generated PV solar power. DL decomposes the multi-step ahead forecasting problem into sub-problems and also uses distributed computing to reduce the computational cost of training a deep neural network and to process big data time series.
2. We conduct a comprehensive evaluation using Australian solar power data for 2 years, measured every 30 min. We evaluate the predictive accuracy of DL and compare it with two state-of-the-art forecasting algorithms: NN and PSF. Our results showed that DL was the most accurate method.
3. We carry out a scalability study to show the suitability of DL for processing large solar power time series, reporting computing times for different time series lengths and comparing DL with NN and PSF.
4. We study the use of multiple input data sources (PV, weather, and weather forecast) and different levels of noise in the weather forecast data. We found that the addition of the weather forecast for the next day to the PV data of the current day improved the accuracy, whereas the addition of the weather data for the current day did not.
5. We analyse how the size of the historical window affects the accuracy of DL. We found that there is no benefit in using more than one previous day.

The rest of the paper is structured as follows. Section 2 reviews of the existing literature related to time series forecasting of solar data. Section 3 introduces the proposed methodology to forecast big data time series. Section 4 describes the data and experimental setup and Section 5 presents and discusses the results. Finally, Section 6 summarizes the main results, providing final conclusions, as well as directions for future work.

2 | RELATED WORK

In this section, we review the recently published approaches for PV solar power prediction, distinguishing between traditional and deep learning techniques.

2.1 | Non-deep learning methods

The non-deep learning methods for time series forecasting can be divided into two groups: classical statistical and data mining techniques (Martínez-Álvarez, Troncoso, Asencio-Cortés, & Riquelme, 2015). With regard to the first group (statistical methods), autoregressive integrated moving average and exponential smoothing have been the most popular methods for predicting PV time series (Dong, Yang, Reindl, & Walsh, 2015; Pedro & Coimbra, 2012). With regard to the second group (data mining methods), neural networks, Support Vector Machines (SVM), and k nearest neighbours have been recently applied to PV solar data. For example, Barbieri et al. (Barbieri, Rajakaruna, & Ghosh, 2017) presented an overview of methods for very short-term PV solar forecasting with cloud modelling. They found that forecasting the irradiance and cell temperature were the best approaches for forecasting PV power fluctuations due to cloud cover, and that a combination of satellite and sky images led to the best results for very-short term forecasting. A neural network, optimized with a genetic algorithm for forecasting the intra-hour PV power, was proposed in Chu et al. (2015). A clustering-based approach based on the weather characteristics was proposed in Wang, Koprinska, and Rana (2017) and Zhang et al. (2018). A survey paper on forecasting methodologies for solar power forecasting was presented in Wan et al. (2015).

Interval forecasts using SVM were studied in Rana, Koprinska, and Agelidis (2015); these type of forecasts were considered as suitable for the highly variable nature of the solar data. A forecasting method based on the weather and power data for the previous days and the weather forecast for the next day was proposed in Z. Wang and Koprinska (2017) for one-day-ahead PV power prediction.

Brecl and Topic (2018) proposed an approach that uses only common weather forecasts, without solar irradiance information, obtaining satisfactory results.

In the last few years, several studies in time series forecasting have focused on creating ensembles of prediction models. Ensembles combine the predictions of several forecasting models and have been shown to be very competitive, and more accurate than single forecasting models in Cerqueira, Torgo, Pinto, and Soares (2017), Koprinska, Rana, Troncoso, and Martínez-Álvarez (2013), and Oliveira and Torgo (2015), including for PV power forecasting (Z. Wang et al. (2017)). Another ensemble method was proposed by Thorey, Chaussin, and Mallet (2018)—an online learning method that generates a weighted combination of PV power forecasts for PV plants located in France; this technique was used to predict solar energy up to 6 days in advance.

2.2 | Deep learning methods

Deep learning methods have gained a lot of interest in recent years due to their excellent results, especially in image and speech recognition tasks (Hinton et al., 2012; Krizhevsky, Sutskever, & Hinton, 2012; LeCun, Bengio, & Hinton, 2015). For surveys on deep learning architectures and applications, see Kamilaris and Prenafeta-Boldú (2018), Mohammadi, Al-Fuqaha, Sorour, and Guizani (2018), and Pouyanfar et al. (2018)

A few recent studies have applied deep learning methods to forecasting tasks, including to energy related time series. For example, Binkowski, Marti, and Donnat (2017) applied deep learning convolutional neural networks (CNNs) and long short-term memory (LSTM) networks to financial and electricity household consumption data with promising results. LSTM networks were also applied for air quality forecasting (Zhou, Chang, Chang, Kao, & Wang, 2019) and indoor temperature prediction (Xu, Chen, Wang, Guo, & Yuan, 2019), and CNNs were used for short-term rainfall prediction (Qiu et al., 2017).

Torres, Fernández, Troncoso, and Martínez-Álvarez (2017) developed a deep learning feed-forward neural network for electricity demand forecasting. The method was used to predict big data times series of Spanish electricity consumption data for 10 years, with a 10-min sampling rate. In Coelho et al. (2017), a deep learning model was applied for household energy demand forecasting, using a GPU parallel architecture for fast processing and model training. A deep learning forecasting model for multi-site PV plant connected with a renewable energy management system was introduced in Lee, Lee, and Kim (2017). Neo, Teo, Woo, Logenthiran, and Sharma (2017) presented an application of Deep Belief NN for forecasting PV solar power.

In Koprinska, Wu, and Wang (2018), CNNs were used for electricity demand and solar power forecasting and were shown to perform similarly to feed-forward neural networks with one hidden layer and to outperform LSTM networks. In Wang et al. (2017), a hybrid method based on wavelet transforms and CNN was applied for PV power forecasting. The wavelet transform was used to decompose the original time series data into several time series with different frequencies; CNNs were then used to extract features from each time series and finally a probabilistic model was applied to forecast each series separately. In Yuchi, Gergely, and Brandt (2018), CNNs were used to correlate PV output to contemporaneous images of the sky and forecast PV power. The effect of the different CNNs and image parameters on the accuracy was also evaluated.

Further, deep recurrent neural networks (RNN) have been shown to provide promising results for predicting PV power in Abdel-Nasser and Mahmoud (2017). Alzahrani, Shamsi, Dagli, and Ferdowsi (2017) used an RNN to forecast the solar irradiance, and compared its performance with several commonly used methods such as SVR and feed-forward neural networks.

After a wide literature review, to the best of our knowledge, we conclude that although there have been previous studies on solar power forecasting using different types of deep learning techniques, none of them deals with big data time series. In this paper, we address this gap by proposing an algorithm for forecasting big solar data using deep learning and evaluating its performance on multiple data sources.

3 | METHODOLOGY

This section presents the proposed methodology to forecast time series, for the context of PV solar data.

The main goal of this work is to predict future values, expressed as $[x_1, \dots, x_h]$, where h is the number of values to predict. The prediction is based on previous values from a historical window w. In this way, the problem can be formulated as:

$$[x_{t+1}, x_{t+2}, \dots, x_{t+h}] = f(x_t, x_{t-1}, \dots, x_{t-(w-1)}), \tag{1}$$

where f refers to the model to be found in the training phase by the algorithm, which will be used to forecast the next h values.

In order to use in-memory data, we utilize Apache Spark cluster-computing. For the deep learning implementation, we choose the H2O machine learning framework, which provides a simple syntax for parallel and distributed programming. However, H2O does not support multi-step forecasting. To deal with this issue, a possible solution is to split the forecasting problem into h forecasting sub-problems. Therefore, it is necessary to compute a prediction model for each sub-problem as follows:

$$x_{t+1} = f_1(x_t, x_{t-1}, \dots, x_{t-(w-1)}), \tag{2}$$

$$x_{t+2} = f_2(x_t, x_{t-1}, \dots, x_{t-(w-1)}), \tag{3}$$

$$\dots \tag{4}$$

$$x_{t+h} = f_h(x_t, x_{t-1}, \dots, x_{t-(w-1)}). \tag{5}$$

From this problem formulation, we can see that each of the *h* values from the prediction horizon is predicted separately, thus removing the propagation error due to previously predicted samples being used to predict the next one. Nevertheless, the computational cost of this methodology is higher than building just one model to predict all *h* values from the prediction horizon because we need to train *h* different models and conduct a hyperparameter search for each of them, instead of training only one model and conducting a single hyperparameter search for optimal parameter selection. The deep learning architecture used for solving each sub-problem is presented in Figure 1.

It is well-known that the values of the hyper-parameters of the deep learning algorithm highly influence the results. To find a good combination of hyper-parameters, we employed the grid search method of H20. The grid-search was used separately for each sub-problem to obtain the best parameter setting as described in detail in Section 5.1.

Figure 2 shows a flow diagram of the proposed methodology. As it can be seen, given a time series data (in column vector format), the task is to find a function that allows to predict a sub-sequence of future values *h* based on the previous know values *w*. This multi-step ahead prediction problem is transformed into *h* sub-problems, where the target value for a sub-problem *i* corresponds to the *i*th value from the prediction horizon. For each of these sub-problems, the data set is divided into training, validation, and test sets. First, the training and validation sets are used for the training and parameter selection. The grid search method computes a model for each combination of hyper-parameters and for each sub-problem. These models are evaluated on the validation set and the best one is chosen to predict the test set and compute the error.

4 | DATA AND EXPERIMENTAL SETUP

4.1 | Data description

We use data from three sources: PV power, weather and weather forecast, for 2 years—from 1 January 2015 to 31 December 2016. This is the same data as in Wang et al. (2017). The PV power is the main data source, but as the generated PV power depends on the weather conditions, we also collected weather and weather forecast data to investigate if its addition can improve the PV power predictions. The three data sets are described below.

PV data. This data set was collected from a rooftop PV plant, located at the University of Queensland in Brisbane, Australia, and is publicly available (http://www.uq.edu.au/solarenergy/). For each day, we only selected the data during the 10-hour daylight period from 7:00 a.m. to

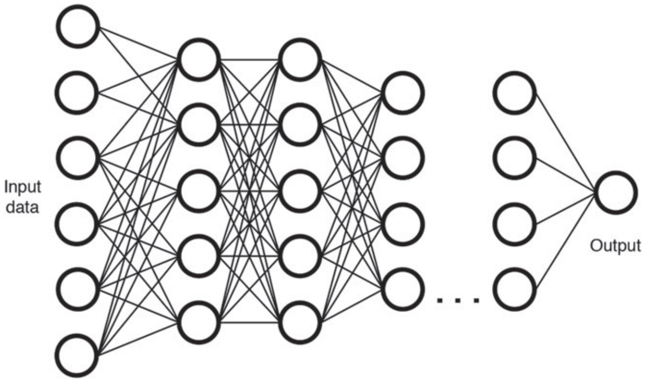


FIGURE 1 DL's architecture

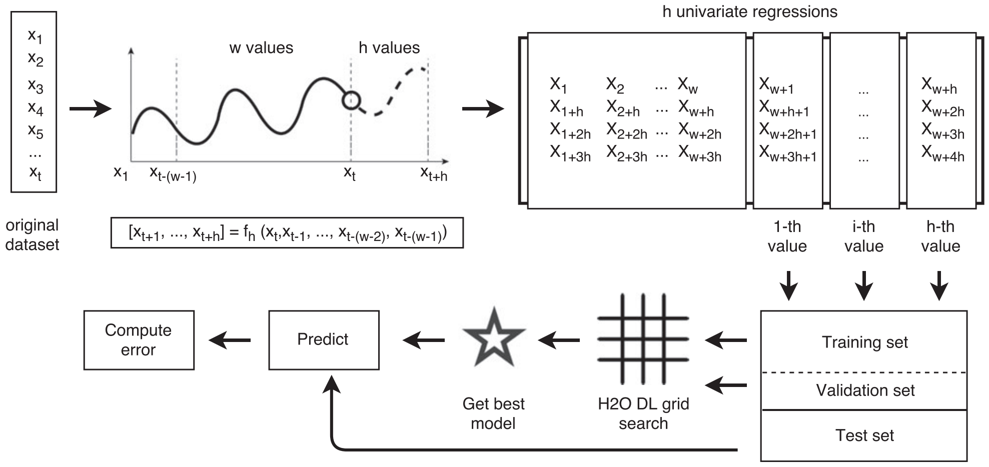


FIGURE 2 Proposed methodology

5:00 p.m. The original PV power data was measured at 1-min intervals and aggregated to 30-min intervals by taking the average value of the interval. As a result, this data set contains 14,620 data points—(365 + 366) days × 20 measurements per day.

Weather data (W). This data set was obtained from the Australian Bureau of Meteorology (http://www.bom.gov.au/). For each day, we collected 14 meteorological variables, described in Table 1. In total, this data set contains 731 days and 14 measurements per day, resulting in 10,234 data points.

Weather forecast data (WF). This data set is a subset of the weather data—it includes four weather variables that are typically available from meteorological bureaus as part of the weather forecast for the next day, as shown in Table 2. Because the weather forecasts were not available retrospectively for 2015 and 2016, we used the actual weather data with added noise at three different levels: 10%, 20%, and 30%. We generated uniformly distributed noise. In total, each of the three versions of this data set contains 2,924 data points—731 days × 4 measurements per day.

Data Preprocessing. There was a small number of missing values—0.82% for the weather data and 0.02% for the PV data. They were replaced using the following nearest neighbour method, applied first to the weather data and then to the PV data: (a) if a day *d* has missing values in its weather vector W^d, we find its nearest neighbour with no missing values, day *s*, using the Euclidean distance and the available values in W^d. The missing values in W^d are replaced with the corresponding values in W^s; (b) if day *d* has missing values in its PV vector P^d, we find its nearest neighbour day *s*, by comparing weather vectors, and then replace the missing values in P^d with the corresponding values in P^s.

The data sets were also re-arranged based on the chosen historical data window and prediction horizon. Specifically, we considered seven historical windows, from 1 to 7 previous days, when predicting the next day. For the PV data, this corresponds to using 20, 40, 60, 80, 100, 120, and 140 past samples as a historical window and 20 samples as a prediction horizon.

All three data sets were normalized to the range of [0,1].

4.2 | Experimental setup

The data was split into training set (the 2015 data) and test set (the 2016 data). The training set was further split into 70% for training and 30% for validation. The training data was used for model training, the validation set was used for parameter tuning, and the test set was used to evaluate the accuracy.

Two performance measures were used to evaluate the accuracy: the mean absolute error (MAE) and the root mean squared error (RMSE). MAE and RMSE are the most commonly used measures for assessing the quality of solar power forecasts (Kostylev & Pavlovski, 2011) and are defined below:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|p_i - a_i|, \tag{6}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - a_i)^2}. \tag{7}$$

TABLE 1 Weather data

ID	Abbreviation	Description
1	DMIN	Daily minimum temperature
2	DMAX	Daily maximum temperature
3	DRAIN	Daily rainfall
4	DSUN	Daily sun hours
5	DMAXWIND	Daily maximum wind speed
6	TEMP9	Temperature at 9:00 a.m.
7	HUM9	Relative humidity at 9:00 a.m.
8	CLOUD9	Cloud cover at 9:00 a.m.
9	WIND9	Wind speed at 9:00 a.m.
10	TEMP3	Temperature at 3:00 p.m.
11	HUM3	Relative humidity at 3:00 p.m.
12	CLOUD3	Cloud cover at 3:00 p.m.
13	WIND3	Wind speed at 3:00 p.m.
14	DSOLARIRR	Daily solar irradiance

TABLE 2 Weather forecast data

ID	Abbreviation	Description
1	DMIN_F	Forecasted daily minimum temperature
2	DMAX_F	Forecasted daily maximum temperature
3	DRAIN_F	Forecasted daily rainfall
4	DSOLARIRR_F	Forecasted daily solar irradiance

All experiments were run on an Intel Core i7-5820 K 3.3 GHz machine with 15 MB of cache, six cores with 12 threads, and 16 GB of RAM memory, working under Ubuntu 16.04 operating system.

5 | RESULTS

This section summarizes the results obtained after applying the proposed deep learning method from Section 3 for forecasting PV solar time series data.

The proposed DL method has been evaluated using a total of seven data sets: (a) PV data alone, (b–d) PV data together with WF data, with three levels of noise in WF, (e–g) PV data together with W and WF data, with three levels of noise in WF. The results are compared with the NN and PSF results from Wang et al. (2017). Section 5.1 presents the optimal parameters obtained by the grid search for each sub-problem. We firstly compare the accuracy and scalability of DL with NN and PSF using only PV data (Section 5.2 and 5.3). Then, we investigate which is the best input data source for DL out of seven data sets, answering four research questions (Q1, Q2, Q3, and Q4) in Section 5.4. We also compare DL with NN and PSF when using W and WF in addition to PV data (Q5) in Section 5.4. Finally, in Section 5.5 we analyse how the size of the historical data window affects the accuracy of the DL method.

5.1 | Parameter selection

As stated before, we applied the grid search strategy available in H2O to find optimal parameters for each sub-problem. Many of the grid search parameters can be customized and are very useful for adapting the network behaviour and improving the training. The following list of parameters were used:

- We varied the number of hidden layers from 1 to 5 and the number of neurons in each layer from 10 to 40.
- The initial weight distribution was set to uniform distribution.
- As an activation function, we chose the hyperbolic tangent function (tanh).
- The distribution function was set to Gaussian distribution.

For each sub-problem of the prediction horizon, an exhaustive search is performed to determine the optimal parameters for the model, using the validation set. When the grid search is completed, the best model for each sub-problem is chosen and used to perform the rest of the experimentation.

Table 3 shows the parameters of the best model obtained for each sub-problem (number of hidden layers and neurons per layer), and also the accuracy (MAE and RMSE) on the training and validation sets. We can see that the best network configuration varied and most often (for 40% of

TABLE 3 Best DL models for each sub-problem

Sub-problem	Hidden layers	Neurons per layer	MAE training	RMSE training	MAE validation	RMSE validation
1	5	39	40.94	58.01	109.13	128.31
2	1	13	62.32	86.83	120.24	145.66
3	3	27	69.57	90.96	132.08	158.33
4	1	37	90.32	120.60	140.98	174.85
5	2	30	98.39	128.22	147.77	184.39
6	2	11	116.47	146.58	162.55	189.90
7	4	14	128.87	161.54	179.44	208.80
8	3	23	134.46	167.14	170.35	212.02
9	2	39	135.11	168.24	177.33	217.07
10	3	32	130.43	161.17	180.26	219.82
11	2	31	134.74	166.59	181.73	218.45
12	5	32	131.25	158.69	174.76	211.29
13	4	37	138.96	165.03	168.33	202.01
14	3	17	138.59	165.03	184.85	213.21
15	5	14	127.95	155.30	167.23	196.42
16	1	39	107.20	132.54	155.12	184.21
17	5	38	92.98	117.94	130.06	152.45
18	4	34	65.72	86.55	100.04	122.07
19	4	40	53.33	74.16	79.49	96.01
20	3	28	48.37	63.70	45.80	57.09

the sub-problems) included three hidden layers, with number of neurons in these layers between 17 and 32. We can also see that the training and validation errors followed the same pattern—they increased until step 13–14 from the prediction horizon (sub-problems 13–14), and then decreased. As expected, the error on the validation set was higher than the error on the training set.

5.2 | Accuracy

Once the optimal configuration of DL for each sub-problem is selected, a new run was launched to predict the test set using this configuration. The results are shown in Tables 4 and 5, and Figure 3.

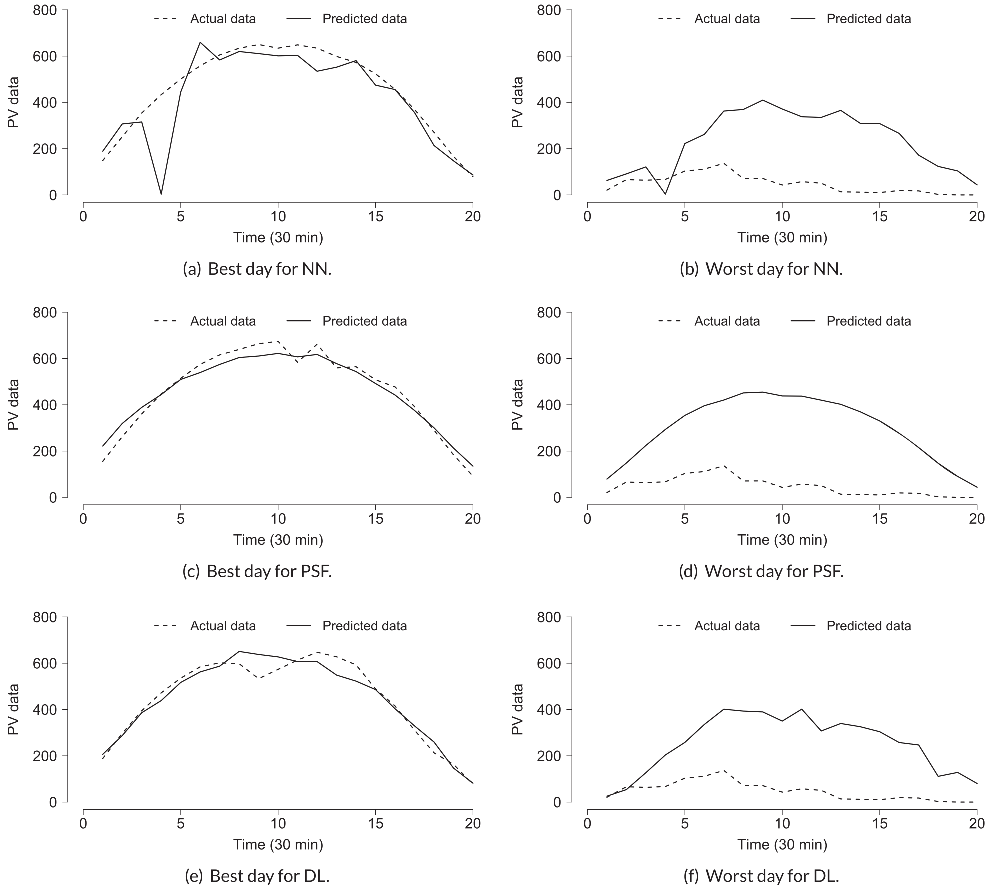
Table 4 shows a comparison of DL with the PSF and NN results from (Wang et al., 2017) where the same data and data split were used. PSF (Martínez-Álvarez et al., 2011) combines clustering with sequence matching. It firstly clusters all days from the training data based on their PV vectors and labels them with the cluster tag. To make a prediction for a new day *d*+1, it extracts a sequence of consecutive days with length w, starting from the previous day *d* and going backwards, and matches the cluster labels of this sequence against the previous days to find a set

TABLE 4 Accuracy of the NN, PSF, and DL algorithms

	NN	PSF	DL
MAE	116.64	119.17	114.76
RMSE	154.16	149.52	148.98

TABLE 5 Best and worst day for NN, PSF, and DL

	Best day		Worst day	
	MAE	RMSE	MAE	RMSE
NN	58.87	106.88	191.52	221.58
PSF	31.72	36.15	252.77	279.12
DL	31.66	41.91	206.33	233.00



(a) Best day for NN.

(b) Worst day for NN.

(c) Best day for PSF.

(d) Worst day for PSF.

(e) Best day for DL.

(f) Worst day for DL.

FIGURE 3 Best and worst day for NN, PSF, and DL algorithms

of equal sequences ES_d. It then follows a nearest neighbour approach—finds the post-sequence day for each sequence in ES_d and averages the PV vectors for these days, to produce the prediction for day *d*+1. The NN model is a multi-layer neural network with one hidden layer (shallow neural network), trained with the Levenberg–Marquardt version of the backpropagation algorithm.

Table 4 shows that DL is the best performing method in terms of both MAE and RMSE. NN is the second best in terms of MAE, and PSF is the second best in terms of RMSE. MAE and RMSE are related measures but RMSE emphasizes less the big differences between the actual and forecasted values.

To study these errors in more detail, we examine the performance of the three methods for the best and worst predicted days. The worst predicted day was the same for all methods (19 June 2016). A further examination revealed that it was indeed an unusual day—there was a heavy rain in central and southern Queensland, causing flash flooding in the roads in Brisbane and more than 9,000 blackouts in the region. This also explains the fact that the average solar power on 19 June 2016 was significantly lower than the one on the same day in other years. On the other hand, the best predicted day was different for the three methods: 7 April 2016 for NN, 3 April 2016 for PSF, and 11 September 2016 for DL. The difference is due to the different nature of the three models.

Figure 3 presents the daily evolution of the actual and forecasted values for the best and worst days, and Table 5 summarizes the daily MAE and RMSE. For the worst day (19 June 2016, the same for the three methods), NN performed best; for the best day (different for every method), DL and PSF were the best performing methods. These results also show that different methods may be more suitable for different days, motivating methods for dynamic selection of the best prediction model for the new day.

5.3 | Scalability

A comparison between the three methods in terms of runtime was also conducted. It includes an evaluation for the original time series, and also for time series 2, 4, 8, 16, 32, and 64 times longer. These longer time series were created from the original by multiplying its length with 2, 4, 8, 16, 32, and 64. The experiments were performed with the optimal DL configurations from Table 3 again.

The results of the scalability analysis are shown in Table 6. As it can be seen, for short time series, the NN and PSF algorithm are faster than DL. However, as the size of the data set increases with a factor of 32 or bigger, the DL method is much faster than the other algorithms. This is because the H2O framework supports distributed and parallel computing, whereas the Matlab implementations of NN and PSF were single-thread.

Figure 4 graphically summarizes the results from Table 6. We can see that the proposed DL model is scalable as its training time increases in a linear way while the training time of the other two methods increases exponentially. This means that the proposed DL approach is highly scalable and is hence suitable for analysing large time series.

5.4 | Use of weather and weather forecast data

The generated PV power depends on the solar irradiance and other meteorological factors. In this section, we investigate if the addition of weather data for the current day (W) and weather forecast data for the next day (WF) can improve the PV power prediction.

The weather and weather forecast data we used have been described in Section 4.1. Recall also that we consider three different versions of the weather forecast data—with 10%, 20%, and 30% noise.

TABLE 6 Computing times (in seconds) for different time series lengths

	×1	×2	×4	×8	×16	×32	×64
NN	0.8020	1.8885	5.4975	24.7970	114.1169	378.0876	2098.0432
PSF	2.4858	14.6286	9.6493	28.9169	101.3701	365.4012	1345.8199
DL	23.0470	23.0480	23.0540	23.0400	22.9600	43.1210	63.2050

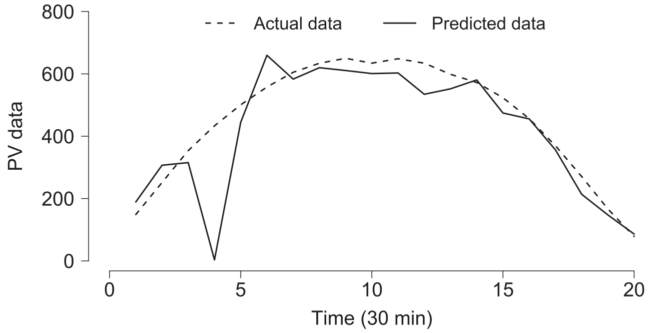


FIGURE 4 Scalability of NN, PSF, and DL algorithms

TABLE 7 Accuracy of the DL for different historical window sizes (from 1 to 7 days)

	PV		PV+W		PV+WF (10%)		PV+WF (20%)		PV+WF (30%)		PV+W+WF (10%)		PV+W+WF (20%)		PV+W+WF (30%)	
Days	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
1	114.76	128.66	126.01	154.00	110.06	135.64	110.27	135.17	109.52	136.32	113.41	140.22	115.32	142.76	122.45	149.83
2	126.15	154.61	129.27	160.44	127.07	156.23	129.28	158.57	123.14	152.34	129.02	158.70	131.24	161.21	135.17	167.08
3	126.03	156.37	133.69	163.97	129.94	160.28	128.66	158.93	128.49	157.83	129.32	160.41	136.93	166.75	133.35	164.48
4	127.77	157.15	131.95	160.80	136.86	167.99	130.51	160.93	132.00	162.55	133.34	164.57	133.61	165.17	131.82	162.43
5	130.74	160.71	130.03	159.64	133.32	162.98	132.64	163.42	129.07	157.71	141.67	173.63	139.93	171.92	139.55	170.35
6	132.02	162.77	133.31	163.87	132.02	162.27	133.74	164.51	136.98	167.57	136.00	166.88	135.78	165.75	142.08	173.77
7	130.66	160.37	136.25	167.07	132.70	163.26	136.83	168.99	134.88	165.99	133.48	163.54	139.97	171.67	137.31	168.90

We investigated the following research questions:

- Q1. Does the addition of the weather data for the current day improve the results?
- Q2. Does the addition of the weather forecast data for the next day improve the results?
- Q3. How does the noise level in the weather forecast data affect the results?
- Q4. Which is the best data source for DL?
- Q5. How does the performance of DL compare with NN and PSF when using weather and weather forecast data, in addition to PV data?

All results are presented in Table 7. Below, we elaborate more on each question and present the relevant results from Table 7 as graphs for visual comparison.

Q1. Using W in addition to PV. We investigate if the addition of the weather data for the current day (W) will improve the prediction. Figure 5 compares the DL's results using the PV data only (PV) and using both the PV and weather data (PV + W). As we can see, the addition of the weather data does not improve the results. A possible explanation for this result is that the weather data is already factored in the PV data as the PV data is highly frequent (every half-hour), and hence, its addition does not contribute any important information for the prediction.

Q2. Using WF in addition to PV. We investigate if the addition of weather forecast data for the next day will improve the performance. Figure 6 shows DL's performance for three different inputs: PV (PV for the current day), PV + WF (PV and weather data for the current day), and PV + W + WF (PV and weather data for the current day, and weather forecast for the next day). In addition, there are three different levels of noise in WF: 10%, 20%, and 30%. Because the noise is only in WF, the results for PV are not affected and are the same for all three noise levels, whereas the results for PV + WF and PV + W + WF change.

We first examine the MAE results. By comparing PV and PV + WF, we can see that DL's performance improves when the weather forecast for the next day is used in addition to the PV data for the current day, and this holds for all three noise levels in WF. By comparing PV + WF and PV + W + WF, we can see that the further addition of the weather data for the current day does not improve the results for all noise levels. Now turning to the RMSE, we observe that RMSE results are consistent with the MAE results, except that the addition of WF does not improve RMSE. This discrepancy between MAE and RMSE shows that we have days with big differences between the actual and predicted values, as RMSE emphasizes such large differences due to the squared term.

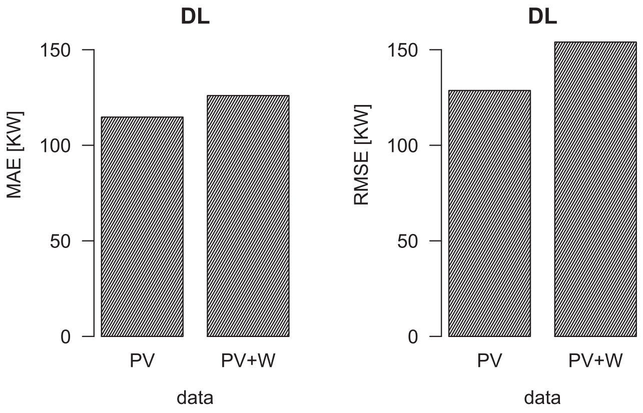


FIGURE 5 Accuracy of DL using PV and PV + W data

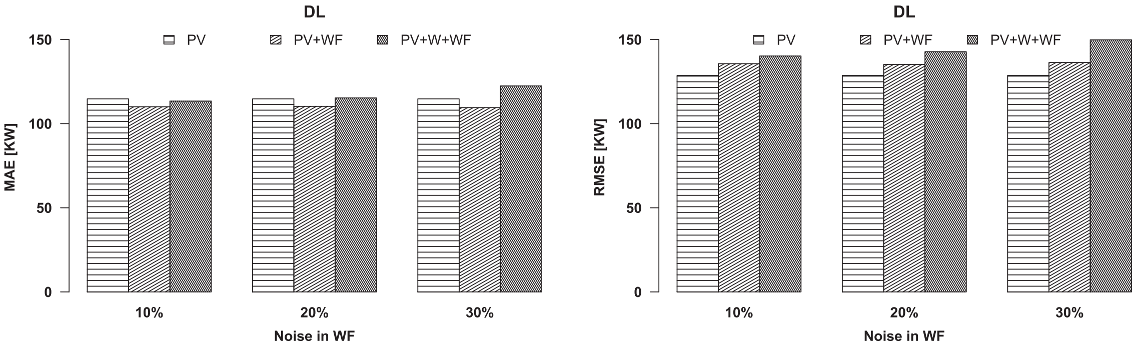


FIGURE 6 Accuracy of DL using PV, PV + WF, and PV + W + WF for three different noise levels in WF

Hence, revisiting Q2 we conclude that the addition of the weather forecast for the next day helps to improve MAE but not RMSE, and that the further addition of the weather data for the current day does not improve the accuracy.

Q3. Effect of the noise level in WF. We investigate the effect of increasing the noise in the weather forecast from 10% to 30% on the predictive accuracy. We first study this effect on the PV + WF data source. Figure 6 shows that the MAE and RMSE results are stable and not affected by the noise level. We now compare the changes in PV + W + WF; we can see that as the noise level increases from 10% to 20%, MAE and RMSE are stable but they increase as the noise increases to 30%. Thus, we conclude that higher level of noise decreases the accuracy of the PV + W + WF data source, whereas the accuracy of PV + WF is not affected.

Q4. Best data source. From Table 7, we can see that DL achieves its best MAE (109.52 kW) when using PV + WF and best RMSE (128.66 kW) when using PV only.

Q5. Comparison of DL with NN and PSF when using W and WF data. We already saw that DL is more accurate than NN and PSF when using the PV data as an input (see Table 4). Here, we assess DL's competitiveness against NN and PFS when using the PV + WF and PV + W + WF data. The NN and PSF methods are implemented as in Wang et al. (2017). Note that the traditional PSF algorithm is univariate and operates on the PV data in our case; to accomodate multivariate data (PV + WF and PV + W + WF), we used the extensions PSF1 and PSF2 (Wang et al., 2017).

Figure 7 presents the results. We can see that for PV + WF, DL is more accurate than NN and PSF, and the advantage increases as the noise level increases. For PV + W + WF, NN is the most accurate method, followed by DL and PSF, and the differences are bigger for MAE than RMSE. We note, however, that DL achieves its best performance while using PV + WF and not PV + W + WF.

Hence, we conclude that DL shows competitive results compared with NN and PSF—it outperforms them on the PV and PV + WF data, and is the second best method on the PV + W + WF data after NN.

5.5 | Historical window size

We investigate how the size of the historical data window *w* affects the accuracy of DL. Table 7 presents the results for *w* varying from 1 to 7 previous days, for all data sources (PV, PV + W, PV + WF, and PV + W + WF) and all three levels of noise in WF. It can be seen that in all cases, the best accuracy is achieved by using only the previous day (day 1 in the table). This is an important observation as it shows that only the data from the previous day is sufficient to make PV power predictions for the next day and that there is no benefit in using more previous days as part of the historical window.

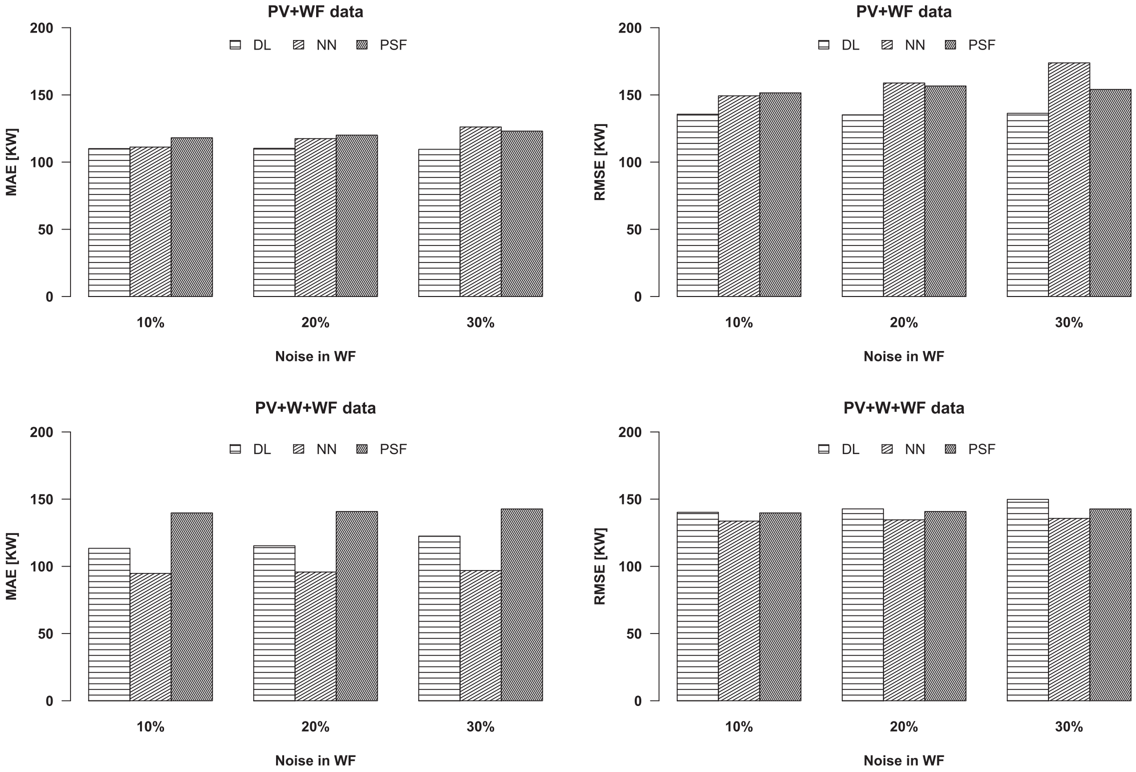


FIGURE 7 Comparison of DL, NN, and PSF using different data sources and noise levels

6 | CONCLUSIONS

In this paper, we introduced DL, a deep neural network approach for predicting the electricity power generated by solar PV systems for the next day.

Our approach has been specifically developed to handle big data time series and has been implemented using the H2O package in conjunction with the Apache Spark cluster-computing framework. It uses a multi-step methodology which decomposes the forecasting problem into several sub-problems, allowing arbitrary prediction horizons. DL was evaluated on Australian data for 2 years and compared with two well-established methods, NN and PSF, demonstrating competitive accuracy results. The scalability analysis demonstrated that DL is suitable for big solar data due to its linear increase in training time, compared with the exponential of NN and PSF. We investigated the use of multiple data sources (PV, weather, and weather forecast) and different levels of noise in the weather forecast. We showed that the addition of the weather forecast for the next day to the PV data for the current day can improve the accuracy, whereas the addition of weather data for the current day is not beneficial. We also studied the effect of the historical window size and showed that there is no benefit in using more than one previous day. In summary, our results show that DL is a promising method for big data solar power forecasting—it scales well and produces competitive accuracy results.

In future work, we plan to develop prediction models for big data based on other types of deep neural networks, for example, LSTM and CNN, and compare them with DL for time series of different nature and length. We will also investigate the application of metaheuristics for more efficient optimization of the hyperparameters of our deep learning network. Other avenues for future work include dynamic selection of the best prediction model for the next day or studying seasonal differences (Koprinska, Rana, & Agelidis, 2011) and building prediction models that are better tuned to the seasonal variations. We also plan to develop dynamic ensembles for big data, motivated by Cerqueira et al. (2017).

ACKNOWLEDGEMENT

The authors would like to thank the Spanish Ministry of Economy and Competitiveness and Junta de Andalucía for the support under projects TIN2017-8888209C2-1-R, TIN2014-55894-C2-R, and P12-TIC-1728, respectively.

ORCID

Alicia Troncoso https://orcid.org/0000-0002-9801-7999

REFERENCES

Abdel-Nasser, M., & Mahmoud, K. (2017). Accurate photovoltaic power forecasting models using deep LSTM-RNN. *Neural Computing and Applications*, 1–14.

Alzahrani, A., Shamsi, P., Dagli, C., & Ferdowsi, M. (2017). Solar irradiance forecasting using deep neural networks. *Procedia Computer Science*, *114*, 304–313.

Barbieri, F., Rajakaruna, S., & Ghosh, A. (2017). Very short-term photovoltaic power forecasting with cloud modeling: A review. *Renewable and Sustainable Energy Reviews*, *75*, 242–263.

Binkowski, M., Marti, G., & Donnat, P. (2017). Autoregressive convolutional neural networks for asynchronous time series. In *Time Series Workshop at International Conference on Machine Learning (ICML)*, Stockholm, Sweden.

Brecl, K., & Topic, M. (2018). Photovoltaics (PV) system energy forecast on the basis of the local weather forecast: Problems, uncertainties and solutions. *Energies*, *11*(5), 1143.

Cerqueira, V., Torgo, L., Pinto, F., & Soares, C. (2017). Arbitrated ensemble for time series forecasting. In *Proceedings of the European Conference on Machine Learning and Principles of Knowledge Discovery in Databases*, Cham, pp. 478–494.

Chu, Y., Urquhart, B., Gohari, S. M. I., Pedro, H. T. C., Kleissl, J., & Coimbra, C. F. M. (2015). Short-term reforecasting of power output from a 48 mwe solar pv plant. *Solar Energy*, *112*, 68–77.

Coelho, I. M., Coelho, V. N., da Luz, E. J. S., Ochi, L. S., Guimarães, F. G., & Rios, E. (2017). A GPU deep learning metaheuristic based model for time series forecasting. *Applied Energy*, *201*, 412–418.

Dong, Z., Yang, D., Reindl, T., & Walsh, W. M. (2015). A novel hybrid approach based on self-organizing maps, support vector regression and particle swarm optimization to forecast solar irradiance. *Energy*, *82*, 570–577.

Ervural, B. C., & Ervural, B. (2018). Improvement of grey prediction models and their usage for energy demand forecasting. *Journal of Intelligent & Fuzzy Systems*, *24*, 2679–2688.

Flannery, T. F., & Sahajwalla, V. (2013). *The critical decade: Australia's future: Solar energy*: Climate Commission Secretariat, Department of Industry, Innovation, Climate Change, Science, Research and Tertiary Education. http://apo.org.au/sites/default/files/docs/ClimateCommission_Australias-Future-Solar-Energy_2013.pdf

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A., Jaitly, N., … Kingsbury, B. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, *29*(6), 82–97.

Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, *147*, 70–90.

Koprinska, I., Rana, M., & Agelidis, V. G. (2011). Yearly and seasonal models for electricity load forecasting. In *International Joint Conference on Neural Networks (IJCNN)*, San Jose, CA, USA, pp. 1474–1481.

Koprinska, I., Rana, M., Troncoso, A., & Martínez-Álvarez, F. (2013). Combining pattern sequence similarity with neural networks for forecasting electricity demand time series. In *Proceedings of the International Joint Conference on Neural Networks*, Dallas, TX, USA, pp. 1–8.

Koprinska, I., Wu, D., & Wang, Z. (2018). Convolutional neural networks for energy time series forecasting. In *International Joint Conference on Neural Networks (IJCNN)*, Rio de Janeiro, Brazil, pp. 1–8.

Kostylev, V., & Pavlovski, A. (2011). Solar power forecasting performance—Towards industry standards. In *First International Workshop on Integration of Solar Power Into Power Systems*, Aarhus, Denmark, pp. 1–11.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, Lake Tahoe, Nevada, pp. 1097–1105.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Lee, J., Lee, I., & Kim, S. (2017). Multi-site photovoltaic power generation forecasts based on deep-learning algorithm. In *2017 International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju, South Korea, pp. 1118–1120.

Livingstone, D. J., Manallack, D. T., & Tetko, I. V. (1997). Data modelling with neural networks: Advantages and limitations. *Journal of Computer-Aided Molecular Design*, *11*, 135–142.

Martínez-Álvarez, F., Troncoso, A., Asencio-Cortés, G., & Riquelme, J. C. (2015). A survey on data mining techniques applied to energy time series forecasting. *Energies*, *8*, 1–32.

Martínez-Álvarez, F., Troncoso, A., Riquelme, J. C., & Aguilar, J. S. (2011). Energy time series forecasting based on pattern sequence similarity. *IEEE Transactions on Knowledge and Data Engineering*, *23*, 1230–1243.

Mohammadi, M., Al-Fuqaha, A., Sorour, S., & Guizani, M. (2018). Deep learning for IoT big data and streaming analytics: A survey. *IEEE Communications Surveys Tutorials*, *20*(4), 2923–2960.

Neo, Y. Q., Teo, T. T., Woo, W. L., Logenthiran, T., & Sharma, A. (2017). Forecasting of photovoltaic power using deep belief network. In *Tencon 2017 - 2017 IEEE Region 10 Conference*, Penang, Malaysia, pp. 1189–1194.

Oliveira, M., & Torgo, L. (2015). Ensembles for time series forecasting. In *Proceedings of the Sixth Asian Conference on Machine Learning*, Nha Trang City, Vietnam, pp. 360–370.

Pedro, H. T. C., & Coimbra, C. F. M. (2012). Assessment of forecasting techniques for solar power production with no exogenous inputs. *Solar Energy*, *86*, 2017–2028.

Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., … Iyengar, S. S. (2018). A survey on deep learning: Algorithms, techniques, and applications. *ACM Computing Surveys*, *51*(5), 92:1–92:36. https://doi.org/10.1145/3234150

Qiu, M., Zhao, P., Zhang, K., Huang, J., Shi, X., Wang, X., & Chu, W. (2017). A short-term rainfall prediction model using multi-task convolutional neural networks. In *2017 IEEE International Conference on Data Mining (ICDM)*, New Orleans, LA, USA, pp. 395–404.

Rana, M., Koprinska, I., & Agelidis, V. G. (2015). 2d-interval forecasts for solar power production. *Solar Energy*, *122*, 191–203.

Reikard, G. (2009). Predicting solar radiation at high resolutions: A comparison of time series forecasts. *Solar Energy*, *83*, 342–349.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, *61*, 85–117.

SolarPowerEurope (2016). Global market outlook for solar power / 2016 - 2020.

Thorey, J., Chaussin, C., & Mallet, V. (2018). Ensemble forecast of photovoltaic power with online crps learning. *International Journal of Forecasting*, *34*(4), 762–773.

Torres, J. F., Fernández, A. M., Troncoso, A., & Martínez-Álvarez, F. (2017). Deep learning-based approach for time series forecasting with application to electricity load. In *Biomedical Applications Based on Natural and Artificial Computing*, Cham, pp. 203–212.

Wan, C., Zhao, J., Song, Y., Xu, Z., Lin, J., & Hu, Z. (2015). Photovoltaic and solar power forecasting for smart grid energy management. *CSEE Journal of Power and Energy Systems*, *1*(1), 38–46.

Wang, Z., & Koprinska, I. (2017). Solar power prediction with data source weighted nearest neighbors. In *Proceedings of the International Joint Conference on Neural Networks*, Anchorage, AK, USA, pp. 1411–1418.

Wang, Z., Koprinska, I., & Rana, M. (2017). Solar power forecasting using pattern sequences. In *Artificial Neural Networks and Machine Learning (ICANN)*, Cham, pp. 486–494.

Wang, Z., Koprinska, I., & Rana, M. (2017). Solar power prediction using weather type pair patterns. In *Proceedings of the International Joint Conference on Neural Networks*, Anchorage, AK, USA, pp. 4259–4266.

Wang, H., Yi, H., Peng, J., Wang, G., Liu, Y., Jiang, H., & Liu, W. (2017). Deterministic and probabilistic forecasting of photovoltaic power based on deep convolutional neural network. *Energy Conversion and Management*, *153*, 409–422.

Xu, C., Chen, H., Wang, J., Guo, Y., & Yuan, Y. (2019). Improving prediction performance for indoor temperature in public buildings based on a novel deep learning method. *Building and Environment*, *148*, 128–135.

Yuchi, S., Gergely, S., & Brandt, B. A. R. (2018). Solar pv output prediction from video streams using convolutional neural networks. *Energy and Environmental Science*, *11*, 1811–1818.

Zhang, X., Li, Y., Lu, S., Hamann, H., Hodge, B. S., & Lehman, B. (2018). A solar time-based analog ensemble method for regional solar power forecasting. *IEEE Transactions on Sustainable Energy*, *10*, 268–279.

Zhou, Y., Chang, F., Chang, L., Kao, I., & Wang, Y. (2019). Explore a deep learning multi-output neural network for regional multi-step ahead air quality forecasts. *Journal of Cleaner Production*, *209*, 134–145.

AUTHOR BIOGRAPHIES

José F. Torres. received the degree in Computer Science from the Pablo de Olavide University, Seville, Spain. He is currently a PhD student in Computer Science at Pablo de Olavide University. His primary areas of interest are big data, data science, deep learning and neural networks, internet of things, time series analysis, and forecasting.

Alicia Troncoso. received the PhD degree in Computer Science from the University of Seville, Spain, in 2005. She was an assistant professor in the Department of Computer Science at the University of Seville from 2002 to 2005. She has been with the Department of

Computer Science at the Pablo de Olavide University since 2005, where she is currently a full professor. Her primary areas of interest are time series forecasting, machine learning and big data.

Irena Koprinska. is an associate professor at the School of Computer Science, University of Sydney, Australia. She holds a PhD in Computer Science and MEd in Higher Education. Her research interests are in neural networks, machine learning, and data mining, both applications and novel algorithms. She also teaches courses in these areas and serves on the programme committee of leading conferences.

Zheng Wang. received a BE degree in Software Engineering with First class Honours from the University of Sydney, Australia, in 2011. He is currently pursuing a PhD degree in the School of Computer Science, University of Sydney. His research interests include neural networks, time series prediction, and feature selection.

Francisco Martínez-Álvarez. received the MSc degree in Telecommunications Engineering from the University of Seville, and the PhD degree in Computer Engineering from the Pablo de Olavide University. He has been with the Department of Computer Science at the Pablo de Olavide University since 2007, where he is currently an associate professor. His primary areas of interest are time series analysis, data mining, and big data analytics.

How to cite this article: Torres JF, Troncoso A, Koprinska I, Wang Z, Martínez-Álvarez F. Big data solar power forecasting based on deep learning and multiple data sources. *Expert Systems*. 2019;e12394. https://doi.org/10.1111/exsy.12394