Energy for Sustainable Development 68 (2022) 1–17

Contents lists available at ScienceDirect

Energy for Sustainable Development

Review

An analysis of different deep learning neural networks for intra-hour solar irradiation forecasting to compute solar photovoltaic generators' energy production

Garazi Etxegarai [a,c], Asier López [a,b], Naiara Aginako [c], Fermín Rodríguez [a,b,*]

[a] Ceit-Basque Research and Technology Alliance (BRTA), Manuel Lardizabal 15, 20018 Donostia/San Sebastián, Spain
[b] Universidad de Navarra, Tecnun, Manuel Lardizabal 13, 20018 Donostia/San Sebastián, Spain
[c] University of the Basque Country, Manuel Lardizabal Ibilbidea 1, 20018 Donostia/San Sebastián, Spain

ARTICLE INFO

Article history:
Received 13 December 2021
Revised 8 February 2022
Accepted 27 February 2022
Available online 15 March 2022

Keywords:
Solar irradiation forecasting
Artificial Neural Network
Very short-term forecasting
Long Short Term memory
Convolutional Neural Network

ABSTRACT

Renewable energies are the alternative that leads to a cleaner generation and a reduction in CO_2 emissions. However, their dependency on weather makes them unreliable. Traditional energy operators need a highly accurate estimation of energy to ensure the appropriate control of the network, since energy generation and demand must be balanced. This paper proposes a forecaster to predict solar irradiation, for very short-term, specifically, in the 10 min ahead. This study develops two tools based on artificial neural networks, namely Long-Short Term Memory neural networks and Convolutional Neural Network. The results demonstrate that the Convolutional Neural Network has a higher accuracy. The tool is tested examining the root mean square error, which was of 52.58 W/m^2 for the testing step. Compared against the benchmark, it has obtained an improvement of 8.16%. Additionally, for the 82% of the tested days it has given a less than 4% error between the predicted and the actual energy generation. Results indicate that the forecaster is accurate enough to be implemented on a photovoltaic generation plan, improving their integration into the electrical grid, not only for providing power but also ancillary services.

Introduction

In recent decades, the energy demand pace is becoming faster than ever (IEA, n.d.). Besides, it is expected to continue, due to the high number of everyday devices that need electricity. Simultaneously, the environmental impact is also increasing. Since the use of traditional energy generation techniques, such as nuclear power, have low efficiency, generate greenhouse effect CO_2 emissions and other pollutants that are transferred to the atmosphere that issue is becoming the centre of governments' attention. This is why some years ago the term "energy transition" began to be heard, to achieve the goal of an utterly renewable generation system. For instance, European Union members signed the 20–20-20 agreement for reducing greenhouse gas emissions by 20%, reducing energy consumption by 20% through greater energy efficiency and increasing the use of renewable sources by 20% (An official website of the European Union, n.d.).

In recent years, one of the most developed renewable energy sources, is the solar energy, along with wind power. In the last years more investment has been made in the development of these two technologies, due to the amount of places with high irradiation and wind possibilities. According to the report presented in 2019 of the International Renewable Energy Agency (IRENA), the price of crystalline PV modules has decreased by 90% between December 2009 and December 2019 (IRENA, n.d.). This is why the use of the photovoltaic energy source has increased considerably. IRENA highlights how the capacity of photovoltaic solar energy increased 14-fold between 2010 and 2019, reaching 580 GW installed by the end of 2019. And the trend is expected to continue to rise in the forthcoming years.

However, the intermittent nature of wind and solar generation increases the need for more flexible and reliable energy generation (Glass & Glass, 2020). Hence, in order to fulfill the requirements of the grid, energy system operators use conventional technologies. Therefore, if a higher

* Corresponding author at: Ceit-Basque Research and Technology Alliance (BRTA), Manuel Lardizabal 15, 20018 Donostia/San Sebastián, Spain.
E-mail addresses:
*garazi.etxegarai@ehu.eus (G. Etxegarai),
*albarriuso@ceit.es (A. López),
*naiara.aginako@ehu.eus (N. Aginako),
*frlalanne@ceit.es (F. Rodríguez).

https://doi.org/10.1016/j.esd.2022.02.002

penetration of renewable generators is desired, a decrease of the unreliability factors must be achieved by the design of accurate forecasters.

Forecasting renewable energies' output power is not a new assignment. Several methods can be found throughout literature. Forecasting is based on predicting future figures within historical databases. A great deal of studies use meteorological parameters, such as solar irradiance, temperature or clear sky index to forecast PV output power (Dong et al., 2014; Gao, Huang, Shi, Tai, Zhang, 2020a; Wang, Xuan, et al., 2020). However, other sources in the literature propose directly forecasting the power generated by renewable energies using previous generated values (Ízgi et al., 2012; Lin & Pai, 2016; Shi et al., 2011).

This study presents two forecasting methods for solar irradiation, based on Artificial Neural Networks (ANN). These tools can forecast solar irradiation for a 10 min prediction horizon. As a consequence, the PV output power could be calculated.

The contributions of this work can be summarised as follows:

- A very short-term solar irradiance forecaster, for within the next 10 min, was developed, in order to integrate into solar photovoltaic generators. Two types of forecaster were designed, which are based on ANN, namely Long Short-Term Memory (LSTM) Neural Network and Convolutional Neural Network (CNN).
- Moreover, RMSE and other error metrics were used to compare not only the developed two forecasters between them, but also with the results provided in the literature.
- Finally, the generated energy is calculated and the deviation between the predicted and actual generated energy is computed. Results show that the tool's accuracy is high enough to integrate on a photovoltaic plant.

The remaining sections of the article are organized as follows: Section 2 classifies the most relevant forecasters. Section 3, describes the database used. In Section 4, different models for the solar irradiation forecasting are analysed, as well as the two proposed models are examined. Section 5, presents the results obtained through proposed models and compares obtained results with those provided in the literature. Finally, Section 6 presents the conclusions of this study.

Irradiation forecasting algorithms

Considering meteorological forecasters, two criteria can be followed to classify them: prediction horizon and the method the forecaster is based on.

Prediction horizon

Prediction horizon is the span of time into the future for which forecast data are to be prepared. Although some researchers propose three categories of the forecast horizon (Raza & Khosravi, 2015), others distinguish four categories (Raza et al., 2016): a) very short-term, predictions of 1 min to several min ahead, b) short-term, predictions of 1 h or several hours ahead to 1 day or 1 week ahead, c) medium-term, predictions of 1 month to several months ahead and d) long-term forecast, predictions of 1 to 10 years ahead. Each prediction horizon involves different purposes (Hyndman & Fan, 2010; Soman et al., 2010). Table 1 summarised them.

Forecaster model

The second criterion to classify forecasting algorithms relies in the forecaster method (Raza et al., 2016; Soman et al., 2010; Tiwari et al., 2018). Here, two groups are distinguished, namely physical and statistical methods.

Physical method

Physical methods are based on a mathematical set of equations, which describe physical states and dynamic motions. Methods like numeric weather prediction (NWP) (Mathiesen & Kleissl, 2011; Tiwari et al., 2018) or sky-imagery (Caldas & Alonso-Sárez, 2019; Dong et al., 2014; Wang, Xuan, et al., 2020) are the most popular ones. Mathiesen & Kleissl (2011) presents the application of different NWP tools to predict global horizontal irradiation (GHI) for a day ahead on the continent of the United States. These techniques present difficulties with time scales smaller than few hours so, generally they are more reliable in long-term forecasting.

Dong et al. (2014) uses satellite image analysis to determine the cloud cover index. Then, using a back propagation multilayer perceptron model they derive the solar irradiance from the cloud cover index. This hybrid method, concluded that for an hourly irradiance forecasting, it has superior forecasting accuracy compared with other popular statistical time series models. Physical methods required a high knowledge of meteorological equations. In addition, the equipment used in those techniques tends to have high costs. Hence, physical methods are frequently rejected due to the required knowledge and equipment's' costs.

Statistical method

Statistical techniques use historical time-series data and real-time data to learn patterns. Therefore, the forecast accuracy of the statistical model depends on the quality and dimensions of the available data. Generally, they have less input data than physical methods, and are easier to model. Besides, they are less expensive in contrast to physical methods. These methods show better performance in short-term forecast than NWP models. Statistical techniques can be divided in two sub-groups (Ahmed et al., 2020; Raza et al., 2016); namely, time series model and artificial intelligence (AI) models.

Time series models, also known as conventional linear statistical techniques, are exponential smoothing, autoregressive moving average (ARMA) and autoregressive integrated moving average (ARIMA) (Belmahdi et al., 2020). Those techniques lead to limitations when forecasting non-linear data, such as solar irradiation. Hence, they are usually combined with machine learning algorithms.

Machine learning algorithms are designed to analyse the input data, implementing learning skills; in this way, logical reasoning is obtained. Wang, Liu, et al. (2020) realised a thorough analysis of the AI methods used for the solar power forecasting. In conclusion, Artificial Neural Networks (ANN) and Support Vector Machine (SVM) are the most

Table 1
Summary of the different prediction horizons and its applications.

Prediction horizon	Range	Applications
Very short-term	1 min to several min ahead	– Power smoothing – Monitoring of real-time electricity dispatch – Electricity marketing/ pricing – PV storage control
Short term	1 h or several hours ahead to 1 day or 1 week ahead	– Economic load dispatch – Power system operation – Control of RE management system – Scheduling
Medium term	predictions of 1 month to several months ahead	– Maintenance scheduling – Energy rationing
Long term	predictions of 1 to 10 years ahead	– Transmission and distribution – Detecting seasonal trends

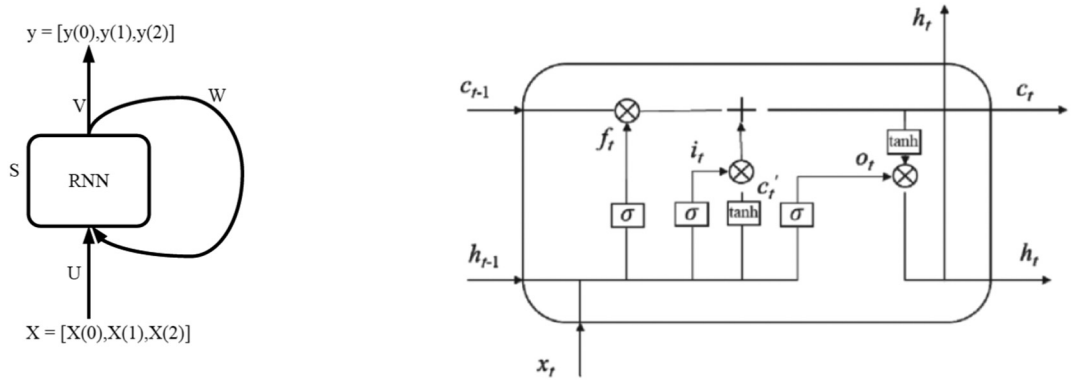

Fig. 1. Schema of a RNN (left) (Wang, Zhang, et al., 2019). Schema of a LSTM cell (right) (Li et al., 2019).

used techniques to forecast the solar power for a very-short term prediction horizon.

A remarkable method that some researchers suggest (Gutierrez-Corea et al., 2016; Mukhoty et al., 2019), is considering the use of nearby location information. Adding the historical spatial and temporal information of surrounding locations, the accuracy of the algorithm can significantly improve. Mukhoty et al. (2019) shows the influence of such method on four different AI algorithms; Feed Forward Neural Network (FFNN), Recurrent Neural Network (RNN), Long Short-Term Neural Network (LSTM) and Gradient Boosted Regression Tree (GBRT), for the forecasting of solar irradiation 1 to 24 h ahead. Mukhoty's model presents the mean absolute error (MAE) and the root mean square error (RMSE) of the four methods individually. After, the spatiotemporal information of the nearby 16 stations is added.

Finally, the above mentioned methods are simple methods, although they can be combined to create hybrid methods (Dong et al., 2014; Kromer et al., 2014; Yan et al., 2019). These methods are intended to achieve better results than simple methods. Nevertheless, it is not always worthwhile, since hybrid methods are more complex and require more training time. Usually these methods achieve better results in long term horizons. For instance, Dong et al. (2014) proposed a hybrid method to forecast solar irradiance one hour ahead. The proposed method, combined satellite image analysis and a hybrid exponential smoothing state space (ESSS) model together with ANN. Firstly, a classification of cloud cover index is created using satellite image analysis and the self-organizing map (SOM). After, the ESSS forecast the next time step cloud cover index. Finally, the ANN model was applied to derive the solar irradiance from the forecasted cloud cover index.

The database

Since the basics of the neural networks are based on learning patterns of the historical data, selecting the database is considerably one of the most important steps. The database needs to be accurate, without lack of data. Proximity is also an important factor when selecting the database. To develop this study the data has been obtained through Euskalmet (Euskalmet, n.d.), a governmental agency of the Basque Country, which provides meteorological data. The historical data for years 2015, 2016 and 2017 has been downloaded. The data is recorded with intervals of 10 min. While database for the years 2015 and 2016 was applied for the training, 2017's data was left for testing. Due to the fact that the neural network has not seen the data from year 2017, the real accuracy of the forecaster can be tested.

The data downloaded from Euskalmet included the subsequent parameters: solar irradiation values, atmospheric pressure, relative air humidity and air temperature. As presented in Rodríguez et al.'s (2018) work, we have only considered the solar irradiation values as inputs for the ANN. Given that only parameters that have a strong correlation in the forecast model output should be considered as inputs. In Rodríguez et al.'s work, the relationship between the irradiance and the remaining parameters was analysed by means of a Pearson correlation. The analysis concluded that the atmospheric pressure, the air relative humidity and the air temperature, depending on the season of the year, presented stronger or weaker relationship. Therefore, they do not have a stable influence on the irradiation. In consequence, they are not taken into account as input for the ANN.

Summarizing, the data given to the training network input is composed of a vector with a length of 146 values. The first value corresponds

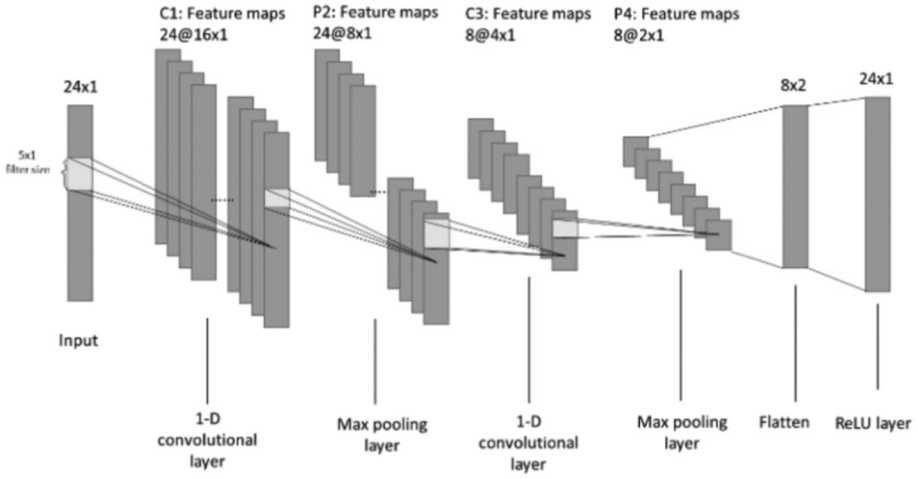

Fig. 2. An example of a CNN (Koprinska et al., 2018).

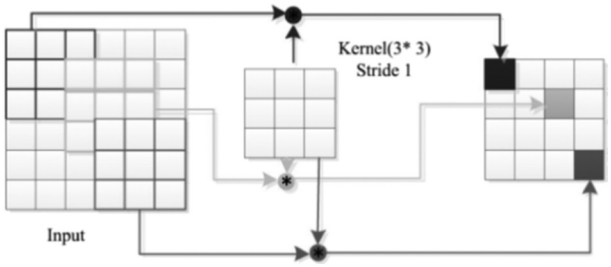

Fig. 3. Convolutional layer process (Wang, Qi, & Liu, 2019c).

to the season of the day being forecasted, the second values determines the time of the day. Finally, the remaining 144 values correspond to the irradiation values recorded every 10 min of the previous 24 h. The output of the ANN is the forecasted irradiation value.

Methods

Analysed forecasting models

After reviewing the literature, it is concluded that the ANNs are appropriate methods to forecast solar irradiation for a very-short term prediction horizon. Hence, LSTM neural networks and CNN will be studied. In addition, the persistence method is also applied, since it is easy to implement and usually is used as benchmark for analysing the improvement of new forecasters.

Persistence method

The concept adopted by this technique is based on the assertion that tomorrow's weather will be the same as today's. Therefore, the prediction obtained by this method is one step back from the real value. This technique usually achieves relatively good values in very short-term horizons. Surprisingly, sometimes the results obtained in this way are better than those obtained by other more complex methods (Ahmed et al., 2020; Soman et al., 2010). That is why it is used as a benchmark, as computationally is the easiest, fastest and least expensive method of all. Hence, any forecast method that is developed is usually tested against the persistence method, to check the improvement over this technique. The equation that describes this forecast model is:

$$P(t+\Delta t) = P(t) \tag{1}$$

where; $P(t+\Delta t)$ is the future predicted value for the forecast horizon Δt, and $P(t)$ is the actual value.

Artificial neural networks

Artificial neural networks are a field of the AI, which digitalizes human abilities. By means of replacing the functioning of the human brain, the performance of the computers has significantly improved, achieving the ability to recognise patterns and predict future unseen values. The artificial neuron is the processing unit. Those artificial neurons are divided into layers, which are communicated by synaptic junctions. Synaptic junctions are responsible of transferring the information between different neurons. All the received information is sum on the artificial neurons. Provided that the information exceeded the cut-off point, it will be sent to the next neuron or to the output (Kalogirou, 2001).

Two types of artificial neural networks are going to be tested, given the potential they present in terms of pattern recognition and data classification. Namely, LSTM and CNN.

Long Short-Term Memory neural networks. LSTM neural networks are a variation of RNN. RNNs have a feedback where the output value, after a time delay, returns to the input, along with the input vector (see Fig. 1 (left)). This feedback loop provides them with a small memory cell, which is updated at every time step. However, with large data, RNNs tend to suffer from the very common problem of vanishing or exploding gradients (Agrawal et al., 2018; Mukhoty et al., 2019). This was the primary encouragement for the development of LSTM. Therefore, LSTM has the ability to remember large data sequences. This capacity is achieved by replacing the original cell by a more complex structure, as can be seen in Fig. 1 (right).

The structure of each LSTM cell is composed of a series of 'gates'. These gates allow data to be deleted, filtered or added to the next cell. Thus, gates allow passing only data that characterizes the output (Ludwing, 2019). These gates are; an input gate, a forget gate, an output gate and a cell state that has long-term memory.

- The input gate allows new data to flow in. Also, it updates the status of the memory cell by means of a mechanism.
- The forget gate decides which information should be rejected, returning 1 or 0.
- The output gate depending on the input data and the state of the memory cell, transmits the information to the next LSTM cell.

The mathematical expressions for each gate are as follow (Cao et al., 2020; Li et al., 2019):

Input gate:

$$\boldsymbol{i_t} = \boldsymbol{S}(\boldsymbol{w_{xi}} \cdot \boldsymbol{x_t} + \boldsymbol{w_{hi}} \cdot \boldsymbol{h_{t-1}} + \boldsymbol{b_i}) \tag{2}$$

Forget gate:

$$\boldsymbol{f_t} = \boldsymbol{S}(\boldsymbol{w_{xf}} \cdot \boldsymbol{x_t} + \boldsymbol{w_{hf}} \cdot \boldsymbol{h_{t-1}} + \boldsymbol{b_f}) \tag{3}$$

Output gate:

$$\boldsymbol{o_t} = \boldsymbol{S}(\boldsymbol{w_{xo}} \cdot \boldsymbol{x_t} + \boldsymbol{w_{ho}} \cdot \boldsymbol{h_{t-1}} + \boldsymbol{b_o}) \tag{4}$$

Cell memory:

$$\boldsymbol{c_t} = \boldsymbol{i_t} \odot \boldsymbol{c_t}' + \boldsymbol{f_t} \odot \boldsymbol{c_{t-1}} \tag{5}$$

$$\boldsymbol{c_t}' = \boldsymbol{\tanh}(\boldsymbol{w_{xc}} \cdot \boldsymbol{x_t} + \boldsymbol{w_{hc}} \cdot \boldsymbol{h_{t-1}} + \boldsymbol{b_c}) \tag{6}$$

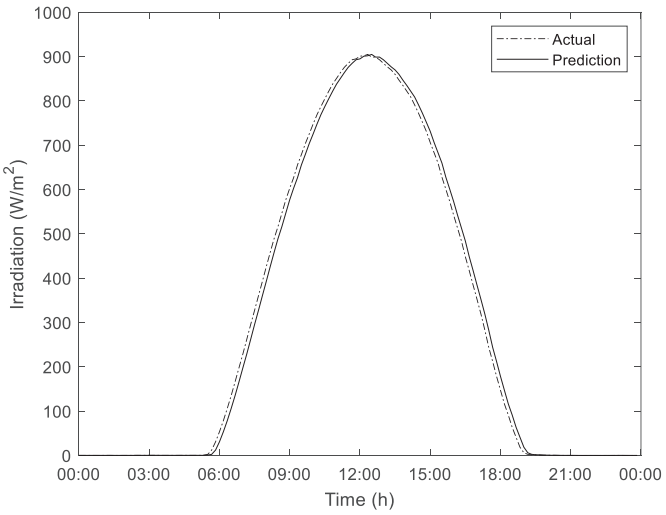

Fig. 4. Persistence models' forecast values vs. real values for August 21, 2017.

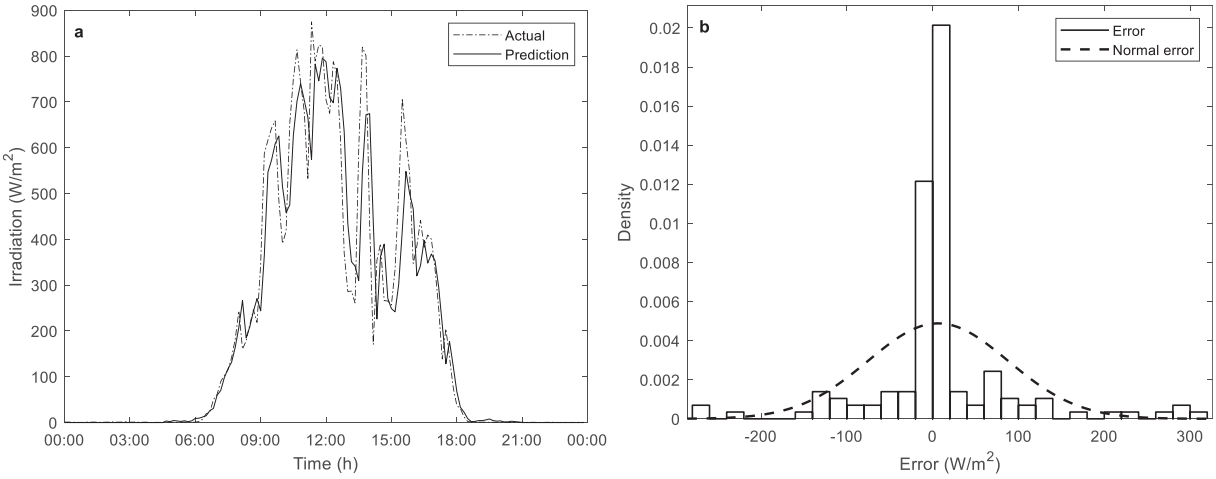

Fig. 5. a) LSTM models' forecast values vs. real values for an overcast day, March 23rd, 2017; b) error histogram and normal distribution.

LSTM cell output:

$$\boldsymbol{h_t} = \boldsymbol{o_t} \cdot \tanh(\boldsymbol{c_t}) \tag{7}$$

where: w_i, w_f, w_o are the input, forget and output weights respectively. b_i, b_f, b_o are the input, forget and output biases. w_c and b_c are the diagonal weights and bias matrices for peephole connections. S and tanh are the activation functions. The sigmoid and tangent functions are often used in LSTM neural networks. Finally, $\odot$ is the element-wise product.

To develop this configuration, some parameters need to be fixed, such as the number of layers, the number of neurons in each layer, the training algorithm and the number of epochs. Finally, another parameter namely 'learn rate' will also be fixed. This helps to get a faster training.

Convolutional neural networks (CNN). In recent years, a great deal of literature has analysed CNN (Feng & Zhang, 2020; Gao, Huang, Shi, Tai, Zhang, 2020b; Koprinska et al., 2018; Ploysuwan, 2019; Wang, Qi, et al., 2019b). Initially, CNNs were designed to take part in Computer Vision tasks. Nevertheless, they have proven to be successful in areas such as object recognition, speech recognition and classification tasks among others. One of the main advantages of CNNs is their ability to learn useful features automatically from high dimensional data without manual feature engineering (Gao, Huang, Shi, Tai, Zhang, 2020b; Koprinska et al., 2018; Ploysuwan, 2019; Wang, Zhang, et al., 2019).

CNNs can be composed of different types of layers, such as convolution layer, pooling layer, batch normalization layer, activation layers and fully connected layers. Depending on the task to be solved, different combinations of layers will be used. An example of different layers combination can be seen in Fig. 2. Nevertheless, to create a CNN at least one convolutional layer needs to be implemented. The convolutional layer is composed of filters, also known as kernels. Those filters passed through all the input data learning and detecting features. A convolutional operation can be seen in Fig. 3. Further information about the meaning of the layers can be found in Refs. (Feng & Zhang, 2020; Wang, Guo, et al., 2020; Wang, Qi, et al., 2019b; Wang, Qi, & Liu, 2019c; Wang, Zhang, et al., 2019).

To design a CNN the following parameters need to be fixed: the layers configuration, the number of layers, each layer parameters, the training algorithm and the number of epochs.

Evaluation error metrics

To assess the performance of the forecasters, several error metrics are used in the literature (Ahmed et al., 2020; Li et al., 2019; Wang, Qi, et al., 2019b; Wang, Qi, & Liu, 2019c; Yi Hong et al., 2020). Although, the most used ones are: root mean square error (RMSE), mean absolute

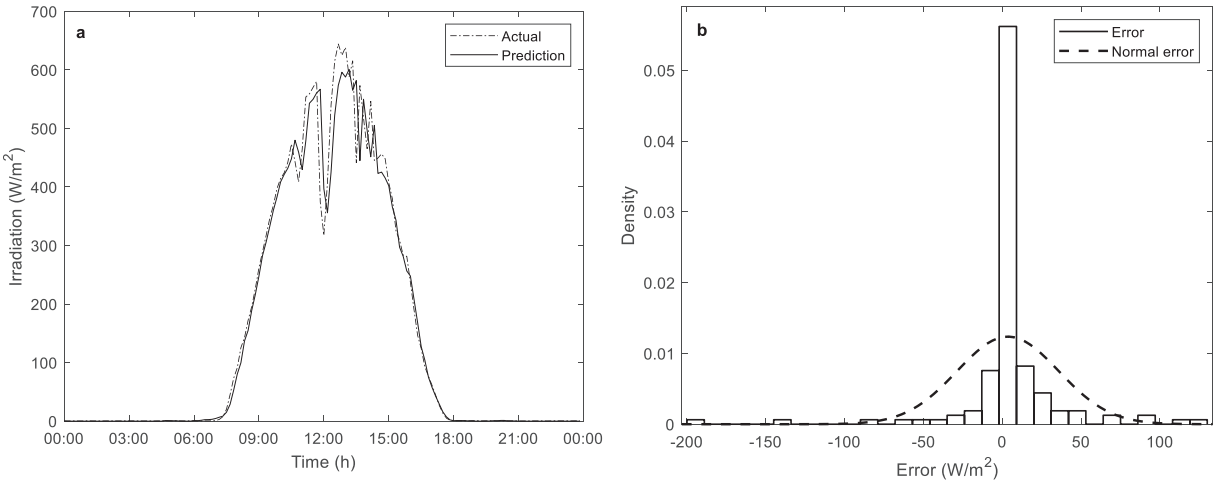

Fig. 6. a) LSTM models' forecast values vs. real values for a partially cloudy day, February 15th, 2017; b) error histogram and normal distribution.

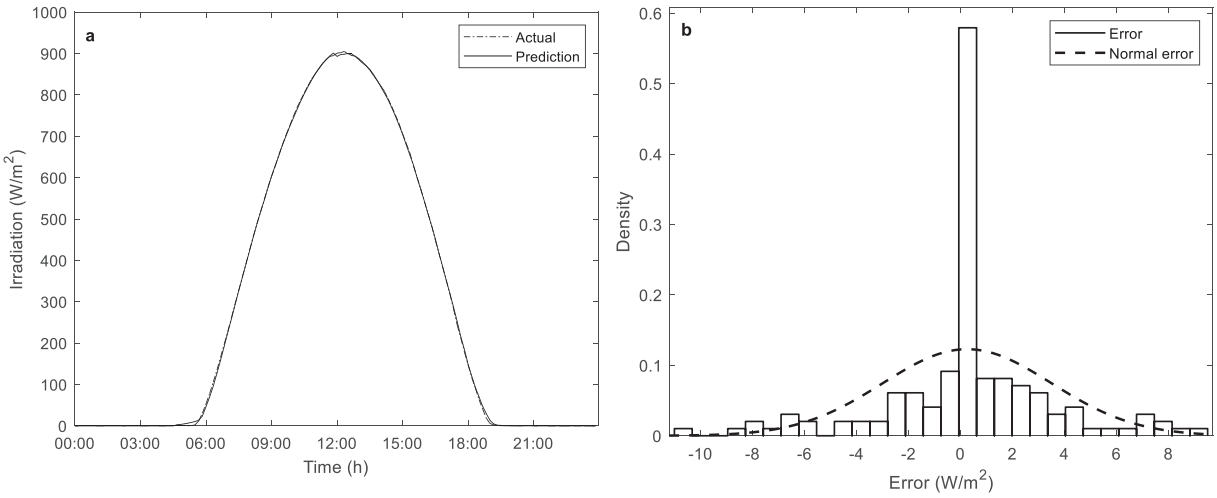

Fig. 7. a) LSTM models' forecast values vs. real values for a sunny day, August 21st, 2017; b) error histogram and normal distribution.

percentage error (MAPE) and mean absolute error (MAE). Those metrics will be useful to compare the developed forecasters between them and against others of the literature.

Even so, these metrics do not give an exact knowledge of the behaviour of the neural network, as a few very high values can ruin the average. Because of that an error histogram will be plotted, so the dispersion of errors can be seen. RMSE and MAE formulas are:

$$\boldsymbol{RMSE} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(Y_i - Y_i'\right)^2} \tag{8}$$

$$\boldsymbol{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|Y_i - Y_i'\right| \tag{9}$$

where; Y_i is the real value, Y_i' is the predicted value and N are the amount of predictions.

Results and discussion

Persistence model

As discussed in Section 4.1.1 the persistence method is used as benchmark. Therefore, persistence model results for our study corresponds to a RMSE for training and testing of 59.47 W/m^2 and 57.25 W/m^2, respectively. Once the RMSE values are known, the percentage improvement of the new forecaster can be calculated, by means of the Eq. (10)

$$\textbf{Performance}\ (\%) = \frac{\boldsymbol{RMSE_{reference}} - \boldsymbol{RMSE_{model}}}{\boldsymbol{RMSE_{reference}}} \cdot \mathbf{100} \tag{10}$$

where, $RMSE_{reference}$ will be the RMSE calculated through the persistence model and the $RMSE_{model}$ the one calculated by the method to be compared.

Fig. 4 represents the forecast for August 21, 2017. The dotted line reflects the actual value and the continuous line the predicted one. As mentioned before, the predicted curve has the same shape as the actual one but with a delayed one time step. The testing RMSE value for this day is 18.26 W/m^2.

LSTM model

As no analytical methodology for fixing the structure of an LSTM was found over the literature, an iterative method was used. The MATLAB® software has been used to develop the LSTM model. Regarding the training solver, two algorithms were tested, namely *adam* and *rmsprop*. Concerning the iterative process for fixing the LSTM parameters, first some variables were held constant and the remaining one was modified. After defining the best value for a variable, another different one was modified. The completely LSTM network optimization can be seen in Appendix A.

With the final LSTM network presented in Table 7 of Appendix A, forecast results are of 54.09 W/m^2 for training and 52.57 W/m^2 for testing. If the performance equation is applied, by means of Eq. (10), an improvement of 9.05% and 8.18% is achieved for training and testing, respectively.

To carry out a deeper analysis, days were classified into three classes: sunny days, partially cloudy days and overcast days. Fig. 5a) represents forecast values for an overcast day, March 23rd, 2017. Considering that an isolated large error can distort the overall RMSE result, an error histogram is plotted with a normal distribution of that error. As can be seen in Fig. 5a), the trend of the prediction line follows the actual line, although there is a small difference between real and predicted values. In addition, if Fig. 5b) is analysed, the major density errors are located inside the bell of the normal distribution. Furthermore, the errors which are outside the bell next to zero are responsible for the deviation that affects the RMSE value.

Regarding Fig. 6a), it shows the prediction that the ANN made on February 15th, 2017. Compared against the prediction made for March 23rd, it can clearly be seen how both, the number of sun hours and the maximum irradiation has decrease, since it is a winter day. Hence,

Table 2
RMSE of different types of days by LSTM method and persistence method.

Type of day	Date	RMSE (W/m^2) LSTM	RMSE (W/m^2) Persistence	Improvement (%)
Sunny	06/01/2017	5.65	11.42	50.52
	08/01/2017	9.54	14.26	33.1
	07/04/2017	10.92	20.10	45.67
	09/04/2017	4.70	18.19	74.16
Partially cloudy	04/01/2017	15.61	18.67	16.4
	03/04/2017	32.99	37.84	12.82
	03/05/2017	41.48	48.35	14.21
	10/06/2017	37.00	41.82	11.53
Cloudy	09/02/2017	43.92	45.55	3.58
	02/03/2017	59.15	63.49	6.84
	06/04/2017	71.12	69.74	−1.98
	02/06/2017	82.43	78.19	−5.42

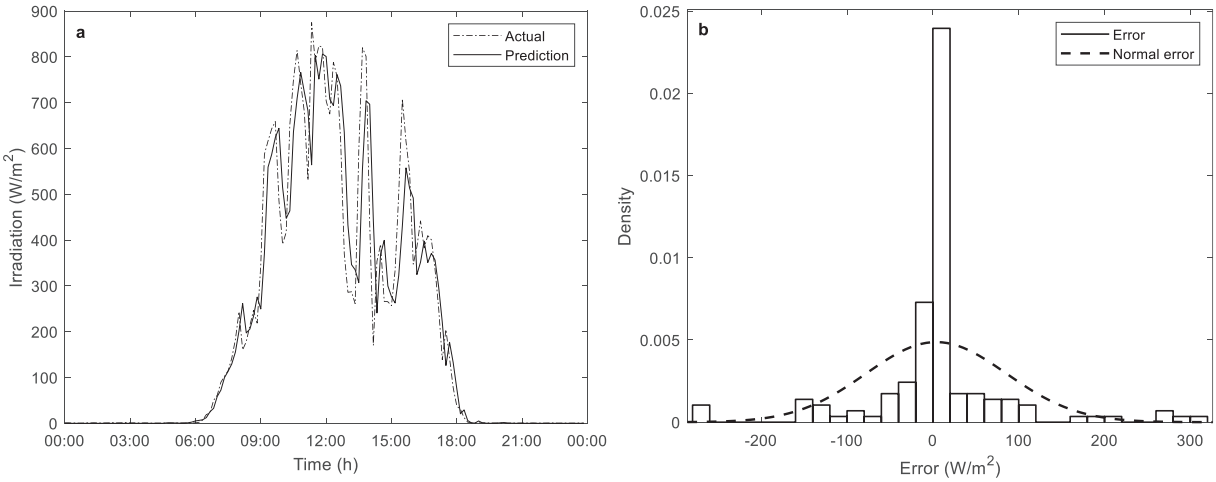

Fig. 8. a) CNN models' forecast values vs. real values for an overcast day, March 23rd, 2017; b) error histogram and normal distribution.

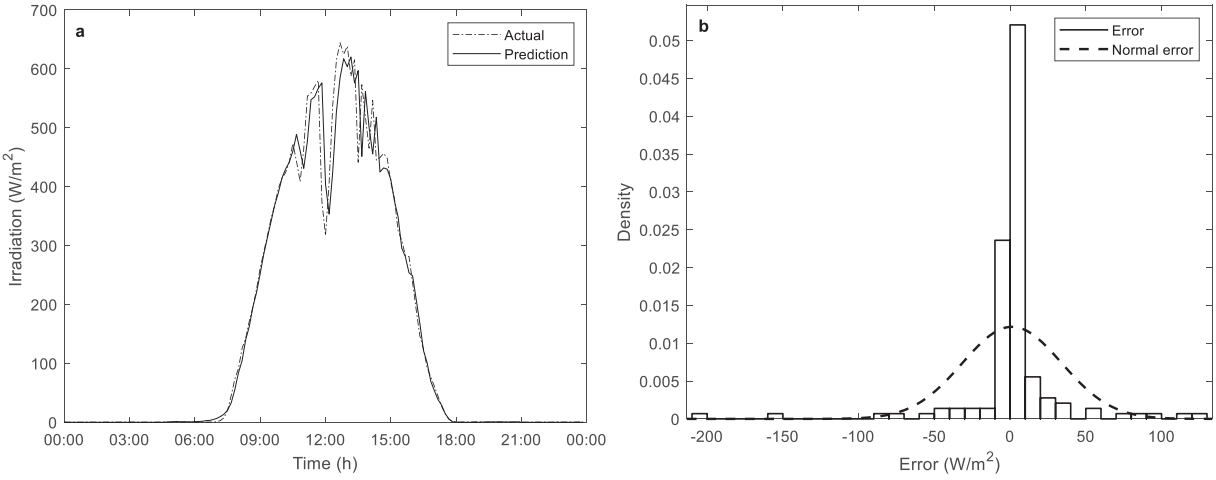

Fig. 9. a) CNN models' forecast values vs. real values for a partially cloudy day, February 15th, 2017; b) error histogram and normal distribution.

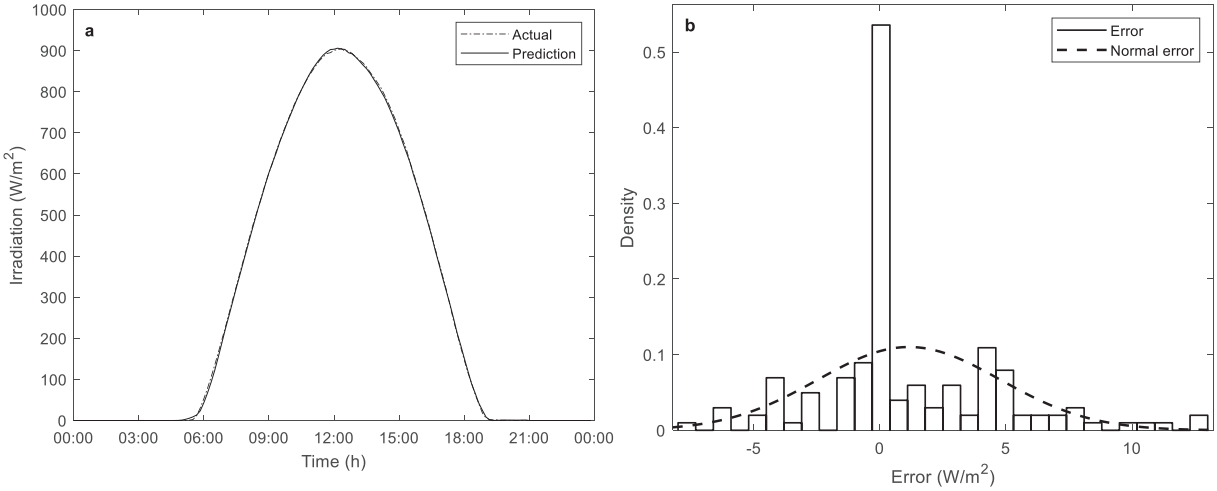

Fig. 10. a) CNN models' forecast values vs. real values for a sunny day, August 21st, 2017; b) error histogram and normal distribution.

Table 3
RMSE of different types of days by CNN method and persistence method.

Type of day	Date	RMSE (W/m^2) CNN	RMSE (W/m^2) Persistence	Improvement (%)
Sunny	06/01/2017	5.07	11.42	55.6
	08/01/2017	9.73	14.26	31.77
	07/04/2017	9.45	20.10	52.98
	09/04/2017	3.92	18.19	78.45
Partially cloudy	04/01/2017	15.75	18.67	15.64
	03/04/2017	32.41	37.84	14.35
	03/05/2017	43.22	48.35	10.61
	10/06/2017	37.90	41.82	9.37
Cloudy	09/02/2017	42.68	45.55	6.3
	02/03/2017	59.19	63.49	6.77
	06/04/2017	70.23	69.74	−0.70
	02/06/2017	83.10	78.19	−6.28

it can be deduced that the ANN works properly. If we analyse Fig. 6b), as in the previous case, the errors which are outside the normal distribution bell are close to zero.

Finally, Fig. 7a) shows the forecast values for a sunny day, August 21st, 2017. We can observe how both the predicted and actual values overlap. The RMSE that the LSTM network has on this day is 4.47 W/m^2. We can conclude that in sunny days the accuracy of the tool is higher. Hence, the less sudden changes, the better performance in the ANN. The improvement for this day is of 75.52%.

Table 2, analyzes a few more days. If we observe the RMSE values, it is demonstrated once again how cloudiness affects the outcome directly. Hence, for higher cloudiness, worse results.

CNN model

As mentioned in Section 4.1.2, a CNN can be composed of different types of layers. Therefore, to determine the most appropriate structure to forecast the solar irradiation, a research through the literature has been done (Chen et al., 2019; Dong et al., 2020; Feng & Zhang, 2020; Gao, Huang, Shi, Tai, Zhang, 2020b; Koprinska et al., 2018; Ryu et al., 2019; Wang, Guo, et al., 2020; Wang, Qi, et al., 2019b; Wang, Qi, & Liu, 2019c; Zang et al., 2018; Zang, Cheng, et al., 2020; Zang, Liu, et al., 2020; Zhoa et al., 2020). Even so, a clear pattern in the structures was not observed. Because of that, an iterative process was done using MATAB®.

Regarding the iterative process, first a variety of structures were tested, concluding that the most appropriate combination was composed by 9 layers: input layer, sequence folding layer, convolution layer, ReLu layer, pooling layer, sequence unfolding layer and an output

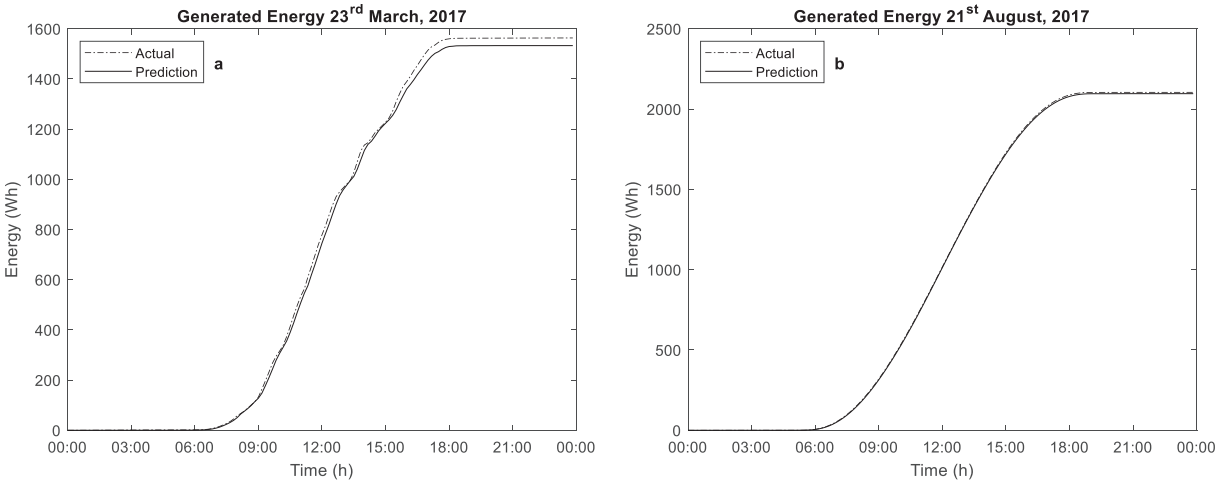

Fig. 11. Real vs. Forecasted energy for an overcast day, 23rd March (left) and real vs forecasted energy for a sunny day, 21st August (right).

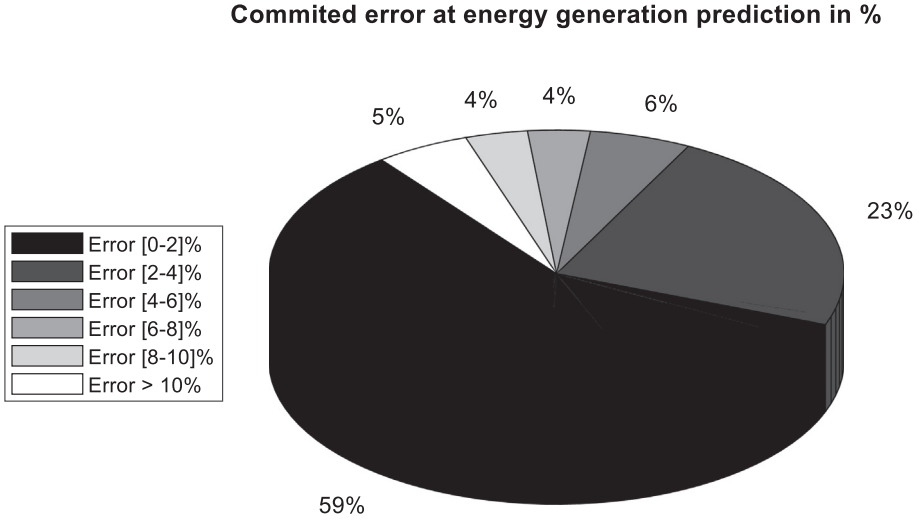

Fig. 12. Distribution error from January to August by the CNN method.

Table 4
RMSE and MAE performance metrics of the CNN forecaster.

Type of day	Date	RMSE (W/m^2) CNN	MAE (W/m^2) CNN
Sunny	06/01/2017	5.07	2.75
	08/01/2017	9.73	3.40
	07/04/2017	9.45	3.92
	09/04/2017	3.92	2.67
Partially cloudy	04/01/2017	15.75	6.02
	03/04/2017	32.41	12.23
	03/05/2017	43.22	17.95
	10/06/2017	37.90	18.36
Cloudy	09/02/2017	42.68	19.71
	02/03/2017	59.19	19.00
	06/04/2017	70.23	33.37
	02/06/2017	83.10	36.70

layer. Afterwards, as in the LSTM forecaster, the training algorithm, the number of epoch and the learning rate were fixed. In Appendix B the completely CNN optimization is presented.

Computing the final structure of Table 15 proposed in Appendix B, the results for the CNN forecaster compared against the results from the persistence model are as follow; they improve from 59.47 W/m^2 to 54.61 W/m^2 in training and from 57.25 W/m^2 to 52.58 W/m^2 in testing. This represents an accuracy improvement of 8.17% and 8.16% in learning and testing steps, respectively.

A deeper assessment is carried out, analysing the performance of three types of days. Fig. 8a) shows the forecasted values for an overcast day, March 23rd, 2017. As more sudden cloudiness changes occur in this day, a bigger difference is given between the predicted and the actual values. Nevertheless, Fig. 8b) shows how the vast majority of the errors are inside the normal distribution bell. Those that are outside are the ones responsible for the distortion that affects the RMSE value, which is 81.72 W/m^2. Although it is a high value, the fact that the more sun hours there are during a day, the more mistakes will be made has to be taken into account. In addition, we can confirm the more periods with clouds a day has, the higher the RMSE will be.

Fig. 9a), presents the values for a partially cloudy day, February 15th, 2017. Thus, this day corresponds to the winter season; the sun hours during the day have decreased. Hence, it can be deduced that the CNN works as expected. The RMSE value for this day is of 32.72 W/m^2, which has also decreased. Additionally, the error histogram of Fig. 9b) performs properly, since the errors are inside the normal distribution bell.

Finally, Fig. 10a), represents the forecasted values for a sunny day. In this case, the predicted values overlap the actual values. The RMSE that the CNN network has on this day is 4.26 W/m^2. This is an improvement of 76.67%. As mentioned, the less cloud changes, the better the performance obtained.

In Table 3 a few more days are analysed. Once again, how cloudiness directly affects the performance of the neural network is confirmed. In addition, on the last two days of the table, the persistence method achieves better results. It can be concluded how sometimes the simplest method performs significantly well.

If both methods are compared, it can be said that CNN outperform slightly the LSTM network. Particularly, in sunny days the CNN has a higher improvement. Whereas, cloudy and partially cloudy days perform similarly. Hence, the CNN will be used to carry out the energy generation analysis.

Analysis of the solar photovoltaic energy generation

Considering that the purpose of this work is to have a prediction of the energy generated by the photovoltaic source, the equation proposed by Rodríguez et al. (2018) has been applied to compute solar PV output power through meteorological parameters.

$$P_{PV}(t) = \eta \cdot S \cdot I(t) \cdot (1 - 0.005 \cdot (T(t) - 25)) \quad (11)$$

where, (t) is the prediction horizon, P_{PV} is the photovoltaic output power [W], η corresponds to the generator efficiency [%], S is the area of the PV generator [m^2], I(t) is the forecasted solar irradiation for the moment t [w/m^2], and T(t) is the ambient temperature value. The influence of temperature on photovoltaic generation is smaller than the irradiation. For this reason, although it is not the ideal case, for this calculation the real values of the temperature will be taken. The values of a commercial panel provided by Rodríguez et al. (Rodríguez et al., 2018) will be used; where, $\eta = 17,59\%$ and $S = 1,6767\ m^2$.

Fig. 11 shows the real accumulated power curve against the forecasted one, for two different days. Whilst the left figure represents the generated energy for an overcast day, the right one shows the energy generated during a sunny day. Corresponding the energy generated for 23rd of March, a difference of 30.55 Wh is given between the calculated and the actual energy, which corresponds to a deviation of 1.95%. Fig. 11 (right) represents the energy generation for the 21st of August. In the case of this summer day, the error arisen is of 0.39%. It can be concluded that cloudiness directly affects the accuracy of the tool.

Finally, Fig. 12 shows the percentage deviation between actual and predicted energy, for months from January to August, both included. To ensure the accuracy of the tool, the results demonstrate that in the 59% of the cases tested, an error between the real and the forecasted energy of less than a 2% is obtained. The mean value for this sample time is of 2.75% error. Although, some few cases are above a 10% error, due to the sudden changes on the cloudiness, it can be said that the forecaster is accurate enough.

Method comparison

As a mean to compare the developed CNN forecaster against others through the literature (Das, 2020; Feng & Zhang, 2020; Mukhoty et al., 2019; Rodríguez et al., 2018; Wang, Xuan, et al., 2020) an extra calculation was made. Table 4 shows the RMSE and MAE performance error metrics, for different degrees of cloudiness.

For instance, Rodríguez et al.'s (2018) works also suggest the use of an artificial neural network, for the development of a solar irradiation forecaster, in particular, a RNN. If the results obtained in Rodríguez et al.'s work are analysed, it can be concluded that the developed CNN network has a higher accuracy for sunny days. While the performance for partially cloudy and overcast days are fairly similar. Regarding the error percentage in the energy generation, again both ANN behave slightly similar. However the error mean value in Rodríguez et al.'s work is of 2.81%, and the one of the CNN is of 2.75%. The smallest error range, which reflects errors under the 2%, in the case of the forecaster developed in this study is bigger than in the Rodríguez et al.'s work. Nevertheless, a deeper analysis should be carried out to determine which ANN has a better performance.

Subhra Das (Das, 2020) suggested using a statistical model, which is based on the anisotropic model of Klucher's, to estimate solar irradiation. For 15 min ahead and 30 min ahead. Also, Subhra Das' work analyses three different days, each one from a different season. Although, if the solar irradiation curves are seen, it can be clearly deduced that the three days correspond to a sunny day. For a 15 min prediction horizon, Subhra Das works RMSEs range from 11.2 W/m^2 to 8.9 W/m^2, while the MAE range from 8.9 W/m^2 to 6.7 W/m^2. If those results are compared against the RMSE range and MAE range expressed in Table 4, regarding sunny days, it can be concluded that the developed CNN outperformed Subhra Das works.

Wang, Xuan, et al. (2020) proposed a physical forecaster, based on real-time surface irradiance mapping model, to predict solar irradiation 1 min ahead to 10 min ahead. The proposed model combines the physical method, to extract the sky image irradiance mapping, and the ANN to learn the relationship between the sky image mapping and the solar

irradiance values. Wang et al. research also divides the days depending on the cloudiness. If the results for an overcast day are analysed, the Wang et al.'s works obtain a RMSE of 120.78 W/m^2 and 120.85 W/m^2. Comparing those values against the ones of Table 4, where RMSEs ranges from 83.10 W/m^2 to 42.68 W/m^2, it can be concluded that the CNN forecaster performs better for overcast days. Regarding partially cloudy days, Wang et al.'s work obtains a RMSE of 92.72 W/m^2 and 98.17 W/m^2, contrasting against the values of Table 4, where the RMSE ranges from 15.75 W/m^2 to 43.22 W/m^2. For this types of days, also, the CNN performs better. Eventually, it is relevant to detail that Wang et al.'s work does not analyse the behaviour of the model for sunny days. Hence the comparison cannot be realised for all the samples days.

Conclusions

The goal of this study was to introduce a tool which is capable of forecasting solar irradiation in the very short-term, specifically 10 min ahead, with the purpose of reducing the uncertainty of the photovoltaic energy generation. The accuracy of the forecaster was validated with real data collected from a station of Vitoria- Gasteiz, Basque Country. For training, data from years 2015 and 2016 was used; While for testing, data for the year 2017. The main conclusions that can be drawn from the study are:

- Two different ANN were designed, namely LSTM and CNN. After comparing the results of both tools it can be deduced that the CNN performance is slightly better. This model provided training and testing RMSEs of 54.61 W/m^2 and 52.58 W/m^2, respectively. Compared against the benchmark persistence model, an accuracy improvement of 8.17% and 8.16% is obtained in training and testing.
- The CNN forecaster developed in this study is compared against another ANN, such as a RNN. It outperforms forecasting on sunny days. Although when partially cloudy and overcast days are forecast both tools behave slightly similar. Therefore, in future studies the combination of two forecasters can be analysed. For instance, the CNN network could be used for sunny days, while another type of forecaster would predict cloudy and partially cloudy days.
- Percentage distribution error calculated from January to August by the CNN method demonstrates a mean error value of 2.75%, with an error rate under the 2% in the 59% of the sample days. Furthermore, Figs. 6a, 7a and 8a demonstrate that the trend of the prediction line follows the actual line at every moment. Therefore, it can be concluded that the tools accuracy is high enough to be introduced in a photovoltaic generation plant to reduce their energy production uncertainty.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

Tables 5 and 6 shows the complete optimization process of the LSTM network. Firstly, in Table 5 a small sample of the whole database is used to make the firsts approximations, where, months of winter and summer seasons are used. Between the first tests three blocks can be distinguished. In the first one, using the *adam training solver* different values of neurons are tested. While, in the second block is used the *rmsprop solver*. Finally, the third block tries different epoch values. For those tests can be concluded that in winter season higher values of neurons reach better RMSE values, whereas in summer lower values perform better. However, higher values of neurons take more computational time, for this reason smaller neurons values are going to be used. Regarding the number of epochs, block 3 results of Table 5, show that 40 epoch reaches the lower RMSE value.

Table 5
Optimisation process of the LSTM forecaster, for winter and summer samples.

	Test	Epochs	Neurons	Training Solver	RMSE Training *Winter*	RMSE Testing *Winter*	RMSE Training *Summer*	RMSE Testing *Summer*
Block 1	**1**	250	20	adam	20,10	44,91	61,58	**65,55**
	2	250	40	adam	19,70	44,61	60,26	67,16
	3	250	60	adam	19,62	44,48	**59,61**	67,29
	4	250	200	adam	19,49	**43,55**	60,26	67,33
	5	250	400	adam	**19,45**	44,27	59,75	66,93
Block 2	**6**	250	20	rmsprop	20,80	44,12	63,77	**65,87**
	7	250	40	rmsprop	21,29	43,31	64,28	66,13
	8	250	60	rmsprop	20,97	44,02	**63,45**	66,05
	9	250	200	rmsprop	20,85	43,34	64,35	66,86
	10	250	400	rmsprop	**20,74**	**42,73**	63,77	66,57
Block 3	**11**	20	20	adam	23,56	43,61	–	–
	12	40	20	adam	22,49	**43,54**	–	–
	13	100	20	adam	22,14	44,02	–	–
	1	250	20	adam	20,10	44,91	–	–

Table 6 collects the remaining tests for the completely database. Moreover, an additional parameter is fixed, namely 'learning rate'. Learning rate helps reducing computational time and reaching better results. Together this parameter are other two sub-parameters, those are; learn rate drop period and learn rate drop factor. To conclude, the lower RMSE is reached with 20 neurons and with the *adam training solver*. The best result is highlighted in bold. An overview of the final LSTM network structure is shown in both Table 7 and Fig. 13.

Table 6
Optimisation process of the LSTM forecaster, for the entire database.

	Test	Neurons	Training Solver	Learn Rate	Learn Rate Drop Period	Learn Rate Drop Factor	RMSE Training Data	RMSE Testing Data
Block 1	**1**	20	rmsprop	1	–	–	**56,28**	**54,49**
	2	40	rmsprop	1	–	–	56,52	54,61
Block 2	**3**	20	adam	1	–	–	**56,17**	**54,41**
	4	40	adam	1	–	–	56,74	55,05
Block 3	**5**	20	rmsprop	0.1	–	–	54,67	53,29
	6	20	adam	0.1	–	–	**54,29**	**52,84**
Block 4	**7**	20	adam	0.5	–	–	55,74	54,03
	8	20	adam	0.07	–	–	**54,18**	**52,80**
	9	20	adam	0.05	–	–	54,26	52,80
	10	20	adam	0.01	–	–	54,67	52,95
Block 5	**11**	20	adam	0.07	20	0.2	54,12	52,63
	12	**20**	**adam**	**0.07**	**20**	**0.1**	**54,09**	**52,57**
	13	20	adam	0.07	10	0.1	54,29	52,63

Table 7
Overview of the LSTM network.

Network type	Long Short-Term Memory (LSTM) NN
Inputs	146 – season, time range, irradiation
Outputs	1 - irradiation
Number of Layers	3 – input, hidden (LSTM), output
Number of Hidden Neurons	20
Epochs	40
Learning Algorithm	adam
Learn Rate	0.07
Learn Rate Drop Period	20
Learn Rate Drop Factor	0.1
Activation Functions	Sigmoid, tanh, linear

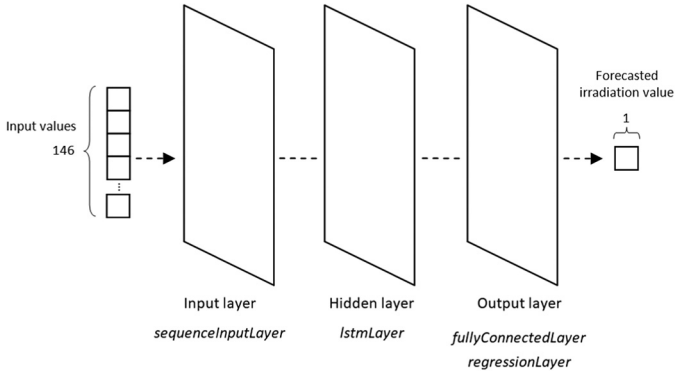

Fig. 13. LSTM network architecture.

Appendix B

As mentioned in Section 4.1.2, a CNN can be composed of different types of layers. Therefore, not only the training parameters have to be fix, also the layers configuration and each one parameter need to be optimised. Hence, the optimisation process will be much longer than the LSTM one. The completely process can be summarised as follow:

I. To begin, a research trough the literature was done making note of the most used layers configurations. Nevertheless, there was not observed a clear pattern in the structures. Therefore, the twenty-nine different configurations presented in Table 8 will be test. The tests will be made with a sample of the whole database, specifically with the winter season, since the computation will be faster. The twenty-nine structures will be tested with the same parameters, to be able to compare them in appropriate circumstances. Table 8 shows obtained results.

II. Then, the best four layers configurations were taken to optimise in detail their layers parameters, such as; the convolution filters size, the number of filters and the pooling window size. These four configurations are highlight in bold in Table 8. The four tables that gather up this parameters optimisation process are Tables 9, 10, 11 and 12.

Table 8
Results of the tested layers configurations.

	Layers configuration	RMSE train	RMSE testing
Test 1	Seq | fold | unfold | regres	**22,63**	**37,94**
Test 2	Seq | fold | unfol | regres	22,46	38,63
Test 3	Seq | fold | unfold | regres	25,61	38,95
Test 4	Seq fold unfold regres	25,46	38,47
Test 5	Seq | fold | unfol | regres	**24,30**	**37,67**
Test 6	Seq | fold | unfol | regres	25,57	39,06
Test 7	Seq | fold | unfold | regres	25,11	38,37
Test 8	Seq | fold | unfold | regres	27,51	44,57
Test 9	Seq | fold | unfold | regres	25,72	42,77
Test 10	Seq | fold | unfold | regres	25,79	42,82
Test 11	Seq | fold | unfold | regres	28,49	47,94
Test 12	Seq | fold | unfold | regres	29,07	42,76
Test 13	Seq | fold | unfold | regres	22,21	40,14
Test 14	Seq | fold | unfold | regres	24,08	39,94
Test 15	Seq | fold | unfold | regres	21,33	39,84
Test 16	Seq | fold | unfold | regres	24,06	42,35
Test 17	Seq | fold | unfold | regres	24,50	38,39
Test 18	Seq | fold | unfold | regres	23,62	38,57
Test 19	Seq | fold | unfold | regres	25,26	39,07
Test 20	Seq | fold | unfold | regres	24,23	40,15
Test 21	Seq | fold | unfold | regres	24,25	38,60
Test 22	Seq | fold | unfold | regres	23,98	39,78
Test 23	Seq | fold | unfold | regres	25,04	39,98
Test 24	Seq | fold | unfold | regres	23,58	39,34
Test 25	Seq | fold | unfold | regres	24,36	37,93
Test 26	Seq | fold | unfold | regres	21,73	39,93
Test 27	Seq | fold | unfold | regres	**22,27**	**38,47**
Test 28	Seq | fold | unfold | regres	**24,35**	**37,94**
Test 29	Seq | fold | unfold | regres	26,01	39,40

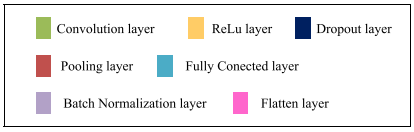

Table 9
Layers parameter optimisation of test 1.

Test 1 Seq | fold | Conv1 | ReLu | Conv2 | ReLu | Flatten | FC | unfold | regres

	Test	Filter size Conv1	Nº filters Conv1	Filter size Conv2	Nº Filter Conv2	RMSE Testing Data
Block 1	**30**	**[3 1]**	**8**	**[3 1]**	**16**	**37,72**
	31	[3 1]	16	[3 1]	32	37,76
	32	[3 1]	32	[3 1]	32	38,33
	33	[3 1]	32	[3 1]	64	38,24
	34	[3 1]	64	[3 1]	64	38,57
	35	[3 1]	64	[3 1]	128	38,69
Block 2	**36**	[5 1]	8	[5 1]	16	38,44
	37	[5 1]	16	[5 1]	32	38,47
	38	[5 1]	32	[5 1]	32	**37,98**
	39	[5 1]	32	[5 1]	64	38,99
	40	[5 1]	64	[5 1]	64	39,09
	41	[5 1]	64	[5 1]	128	39,70
Block 3	**42**	[7 1]	8	[7 1]	16	38,76
	43	[7 1]	16	[7 1]	32	39,29
	44	[7 1]	32	[7 1]	32	40,25
	45	[7 1]	32	[7 1]	64	39,00
	46	[7 1]	64	[7 1]	64	39,91
	47	[7 1]	64	[7 1]	128	39,36

Table 10
Layers parameters optimisation of test 5.

Test 5 Seq | fold | Conv1 | ReLu | Pooling | Conv2 | ReLu | Flatten | FC | unfold | regress

	Test	Filter size Conv1	Nº filters Conv1	Filter size Conv2	Nº Filter Conv2	Pooling Window size	RMSE Testing Data
Block 1	**48**	[3 1]	8	[3 1]	16	[5 1]	38,11
	49	[3 1]	16	[3 1]	32	[5 1]	39,40
	50	[3 1]	32	[3 1]	32	[5 1]	**37,67**
	51	[3 1]	32	[3 1]	64	[5 1]	38,88
	52	[3 1]	64	[3 1]	64	[5 1]	38,91
	53	[3 1]	64	[3 1]	128	[5 1]	38,06
Block 2	**54**	[5 1]	8	[5 1]	16	[5 1]	38,35
	55	[5 1]	16	[5 1]	32	[5 1]	38,61
	56	[5 1]	32	[5 1]	32	[5 1]	38,55
	57	[5 1]	32	[5 1]	64	[5 1]	38,24
	58	[5 1]	64	[5 1]	64	[5 1]	38,55
	59	[5 1]	64	[5 1]	128	[5 1]	**37,66**
Block 3	**60**	[7 1]	8	[7 1]	16	[5 1]	38,41
	61	[7 1]	16	[7 1]	32	[5 1]	38,11
	62	[7 1]	32	[7 1]	32	[5 1]	37,89
	63	[7 1]	32	[7 1]	64	[5 1]	38,04
	64	**[7 1]**	**64**	**[7 1]**	**64**	**[5 1]**	**37,30**
	65	[7 1]	64	[7 1]	128	[5 1]	37,81
Block 4	**66**	[7 1]	64	[7 1]	64	[3 1]	40,14
	64	[7 1]	64	[7 1]	64	[5 1]	**37,30**
	67	[7 1]	64	[7 1]	64	[7 1]	39,92
	68	[7 1]	64	[7 1]	64	[9 1]	38,22

Table 11
Layers parameters optimisation of test 27.

Test 27 Seq fold Conv1 ReLu Conv2 ReLu Conv3 BN ReLu Flatten FC unfold regres

	Test	Filter size Conv1	Nº filters Conv1	Filter size Conv2	Nº Filter Conv2	Filter size Conv3	Nº Filter Conv3	Pooling Window size	RMSE Testing Data
Block 1	**69**	[3 1]	8	[3 1]	16	[3 1]	16	[5 1]	39,18
	70	[3 1]	8	[3 1]	16	[3 1]	32	[5 1]	38,96
	71	[3 1]	16	[3 1]	32	[3 1]	32	[5 1]	38,89
	72	[3 1]	16	[3 1]	32	[3 1]	64	[5 1]	39,20
	73	**[3 1]**	**32**	**[3 1]**	**32**	**[3 1]**	**32**	**[5 1]**	**38,33**
	74	[3 1]	32	[3 1]	32	[3 1]	64	[5 1]	41,06
	75	[3 1]	32	[3 1]	64	[3 1]	64	[5 1]	41,71
	76	[3 1]	32	[3 1]	64	[3 1]	128	[5 1]	40,87
	77	[3 1]	64	[3 1]	64	[3 1]	64	[5 1]	39,56
	78	[3 1]	64	[3 1]	64	[3 1]	128	[5 1]	42,03
	79	[3 1]	64	[3 1]	128	[3 1]	128	[5 1]	39,73
	80	[3 1]	64	[3 1]	128	[3 1]	256	[5 1]	39,41
Block 2	**81**	[5 1]	8	[5 1]	16	[5 1]	16	[5 1]	39,08
	82	[5 1]	8	[5 1]	16	[5 1]	32	[5 1]	39,82
	83	[5 1]	16	[5 1]	32	[5 1]	32	[5 1]	**38,76**
	84	[5 1]	16	[5 1]	32	[5 1]	64	[5 1]	39,80
	85	[5 1]	32	[5 1]	32	[5 1]	32	[5 1]	39,58
	86	[5 1]	32	[5 1]	32	[5 1]	64	[5 1]	39,01
	87	[5 1]	32	[5 1]	64	[5 1]	64	[5 1]	39,58
	88	[5 1]	32	[5 1]	64	[5 1]	128	[5 1]	39,95
	89	[5 1]	64	[5 1]	64	[5 1]	64	[5 1]	39,76
	90	[5 1]	64	[5 1]	64	[5 1]	128	[5 1]	40,12
	91	[5 1]	64	[5 1]	128	[5 1]	128	[5 1]	40,31
	92	[5 1]	64	[5 1]	128	[5 1]	256	[5 1]	39,76
Block 3	**93**	[7 1]	8	[7 1]	16	[7 1]	16	[5 1]	39,96
	94	[7 1]	8	[7 1]	16	[7 1]	32	[5 1]	39,69
	95	[7 1]	16	[7 1]	32	[7 1]	32	[5 1]	40,55
	96	[7 1]	16	[7 1]	32	[7 1]	64	[5 1]	39,92
	97	[7 1]	32	[7 1]	32	[7 1]	32	[5 1]	**39,28**
	98	[7 1]	32	[7 1]	32	[7 1]	64	[5 1]	39,59
	99	[7 1]	32	[7 1]	64	[7 1]	64	[5 1]	40,01
	100	[7 1]	32	[7 1]	64	[7 1]	128	[5 1]	41,25
	101	[7 1]	64	[7 1]	64	[7 1]	64	[5 1]	39,76
	102	[7 1]	64	[7 1]	64	[7 1]	128	[5 1]	41,31
	103	[7 1]	64	[7 1]	128	[7 1]	128	[5 1]	42,17
	104	[7 1]	64	[7 1]	128	[7 1]	256	[5 1]	42,07

Table 12
Layers parameters optimisation of test 28.

Test 28 Seq fold Conv1 ReLu Pooling Conv2 ReLu Conv3 ReLu Flatten FC unfold regres

	Test	Filter size Conv1	Nº filters Conv1	Filter size Conv2	Nº Filter Conv2	Filter size Conv3	Nº Filter Conv3	Pooling Window size	RMSE Testing Data
Block 1	**105**	[3 1]	8	[3 1]	16	[3 1]	16	[5 1]	37,91
	106	[3 1]	8	[3 1]	16	[3 1]	32	[5 1]	38,77
	107	[3 1]	16	[3 1]	32	[3 1]	32	[5 1]	38,52
	108	[3 1]	16	[3 1]	32	[3 1]	64	[5 1]	38,52
	109	[3 1]	32	[3 1]	32	[3 1]	32	[5 1]	38,71
	110	[3 1]	32	[3 1]	32	[3 1]	64	[5 1]	38,76
	111	[3 1]	32	[3 1]	64	[3 1]	64	[5 1]	38,69
	112	[3 1]	32	[3 1]	64	[3 1]	128	[5 1]	38,78
	113	[3 1]	64	[3 1]	64	[3 1]	64	[5 1]	37,92
	114	[3 1]	64	[3 1]	64	[3 1]	128	[5 1]	**37,46**
	115	[3 1]	64	[3 1]	128	[3 1]	128	[5 1]	39,39
	116	[3 1]	64	[3 1]	128	[3 1]	256	[5 1]	38,26
Block 2	**117**	[5 1]	8	[5 1]	16	[5 1]	16	[5 1]	39,09
	118	[5 1]	8	[5 1]	16	[5 1]	32	[5 1]	39,62
	119	[5 1]	16	[5 1]	32	[5 1]	32	[5 1]	38,80
	120	[5 1]	16	[5 1]	32	[5 1]	64	[5 1]	38,91
	121	[5 1]	32	[5 1]	32	[5 1]	32	[5 1]	38,42
	122	[5 1]	32	[5 1]	32	[5 1]	64	[5 1]	38,18
	123	[5 1]	32	[5 1]	64	[5 1]	64	[5 1]	38,94
	124	[5 1]	32	[5 1]	64	[5 1]	128	[5 1]	38,23
	125	[5 1]	64	[5 1]	64	[5 1]	64	[5 1]	37,88
	126	[5 1]	64	[5 1]	64	[5 1]	128	[5 1]	39,95
	127	**[5 1]**	**64**	**[5 1]**	**128**	**[5 1]**	**128**	**[5 1]**	**37,60**
	128	[5 1]	64	[5 1]	128	[5 1]	256	[5 1]	38,40
Block 3	**129**	[7 1]	8	[7 1]	16	[7 1]	16	[5 1]	38,01
	130	[7 1]	8	[7 1]	16	[7 1]	32	[5 1]	38,59
	131	[7 1]	16	[7 1]	32	[7 1]	32	[5 1]	38,00
	132	[7 1]	16	[7 1]	32	[7 1]	64	[5 1]	38,24
	133	[7 1]	32	[7 1]	32	[7 1]	32	[5 1]	39,89
	134	[7 1]	32	[7 1]	32	[7 1]	64	[5 1]	38,19
	135	[7 1]	32	[7 1]	64	[7 1]	64	[5 1]	38,38
	136	[7 1]	32	[7 1]	64	[7 1]	128	[5 1]	**37,78**
	137	[7 1]	64	[7 1]	64	[7 1]	64	[5 1]	38,74
	138	[7 1]	64	[7 1]	64	[7 1]	128	[5 1]	38,63
	139	[7 1]	64	[7 1]	128	[7 1]	128	[5 1]	38,55
	140	[7 1]	64	[7 1]	128	[7 1]	256	[5 1]	38,25

III. After, the best five structures of Tables 9, 10, 11 and 12 were computed with the whole database. Those results can be seen in Table 13.

Table 13
The five best structures of Tables 9, 10, 11 and 12 computed with the entire database.

Test 5 Seq | fold | Conv1 | ReLu | Pooling | Conv2 | ReLu | Flatten | FC | unfold | regress

Test	Filter size Conv1	Nº filters Conv1	Filter size Conv2	Nº Filter Conv2	Pooling Window	RMSE Testing Data	Training time
1	[3 1]	32	[3 1]	32	[5 1]	54,81	~ 5 min
2	[5 1]	64	[5 1]	128	[5 1]	54,46	~ 13 min
3	**[7 1]**	**64**	**[7 1]**	**64**	**[5 1]**	**53,70**	**17 – 19 min**

Test 28 Seq | fold | Conv1 | ReLu | Pooling | Conv2 | ReLu | Conv3 | ReLu | Flatten | FC | unfold | regres

Test	Filter size Conv1	Nº filters Conv1	Filter size Conv2	Nº Filter Conv2	Filter size Conv3	Nº Filter Conv3	Pooling Window	RMSE Tsting Data	Training time
4	[3 1]	64	[3 1]	64	[3 1]	128	[5 1]	54,23	~ 15 min
5	[5 1]	64	[5 1]	128	[3 1]	128	[5 1]	53,67	23 - 27 min

IV. The best result of Table 13 is selected as the final layers configuration. Finally the remaining training solver parameters were fixed, such as, the number of epoch, the training solver and the learn rate factors. Table 14 collects this final optimisation process.

Table 14
Training parameters optimisation process of the final CNN layer structure.

Final structure Seq | fold | Conv1 | ReLu | Pooling | Conv2 | ReLu | Flatten | FC | unfold | regress

	Test	Epoch	Training Solver	Learn Rate	L. R. Drop Period	L. R. Drop Factor	Validation Patience	RMSE Training Data	RMSE Testing Data
Block 1	**3**	5	adam	1	-	-	-	55,75	53,70
	6	10	adam	1	-	-	-	**55,62**	**53,53**
	7	15	adam	1	-	-	-	56,03	54,04
	8	20	adam	1	-	-	-	55,27	53,17
	9	25	adam	1	-	-	-	55,66	53,58
	10	40	adam	1	-	-	-	54,61	**52,53**
Block 2	**11**	40	adam	1	-	-	7	56,96	54,89
	12	40	adam	1	-	-	9	56,26	54,15
	13	40	adam	1	-	-	11	56,18	54,16
	14	40	adam	1	-	-	15	56,76	54,68
	15	40	adam	1	-	-	17	56,29	54,27
BLlock 3	**3**	5	adam	1	-	-	-	55,75	53,70
	6	10	adam	1	-	-	-	**55,62**	**53,53**
	7	15	adam	1	-	-	-	56,03	54,04
	8	20	adam	1	-	-	-	55,27	53,17
	9	25	adam	1	-	-	-	55,66	53,58
	16	5	rmsprop	1	-	-	-	56,38	54,30
	17	10	rmsprop	1	-	-	-	56,42	54,25
	18	15	rmsprop	1	-	-	-	56,25	54,12
Block 4	**19**	10	adam	2	-	-	-	55,66	53,55
	6	10	adam	1	-	-	-	**55,62**	**53,53**
	20	10	adam	0.5	-	-	-	55,76	53,77
	21	10	adam	0.1	-	-	-	55,83	53,66
	22	10	adam	0.07	-	-	-	55,80	53,64
	23	10	adam	0.05	-	-	-	55,71	53,54
	24	10	adam	0.1	-	-	-	57,00	54,68
Block 5	**25**	10	adam	1	3	0.5	-	54,74	52,69
	26	**10**	**adam**	**1**	**3**	**0.2**	-	**54,61**	**52,58**
	27	10	adam	1	3	0.1	-	54,70	52,62
	28	10	adam	1	2	0.5	-	54,78	52,78
	29	10	adam	1	4	0.2	-	54,69	52,64
	30	10	adam	0,5	3	0.5	-	54,92	52,84
	31	10	adam	0,1	3	0.5	-	55,25	53,09

The final CNN forecast structure and its parameters values are summarised in Table 15.

Table 15
Overview of the CNN network.

Network type	Convolutional Neural Network (CNN)
Inputs	146 – season, time range, irradiation
Outputs	1 - irradiation
Number of Layers	9 – input, seq. folding, convolution, ReLu, pooling, convolution, ReLu, seq. unfold, output
Hidden Layers parameters:	
Convolution 1. Filter size	[7 1]
Convolution 1. Number of filters	64
Convolution 2. Filter size	[7 1]
Convolution 2. Number of filters	64
Pooling window size	[5 1]
Epochs	10
Learning Algorithm	adam
Learn Rate	1
Learn Rate Drop Period	3
Learn Rate Drop Factor	0.2

References

Agrawal, R. K., Muchhary, F., & Tripathi, M. M. (2018). Long term load forecasting with hourly predictions based on long-short-term-memory networks. *IEEE Texas Power and Energy Conference (TPEC)*. https://doi.org/10.1109/TPEC.2018.8312088.

Ahmed, R., Sreeram, V., Mishra, Y., & Arif, M. D. (2020). A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization. *Renewable and Sustainable Energy Reviews*, *124*. https://doi.org/10.1016/j.rser.2020.109792.

An official website of the European Union (d). 2020 climate & energy package. https://ec.europa.eu/clima/policies/strategies/2020_en#tab-0-0 (Accessed, February 2021).

Belmahdi, B., Louzazni, M., & El Bouardi, A. (2020). One month-ahead forecasting of mean daily global solar radiation using time series models. *Optik*, *219*. https://doi.org/10.1016/j.ijleo.2020.165207.

Caldas, M., & Alonso-Sárez, R. (2019). Very short-term solar irradiance forecast using all-sky imaging and real-time irradiance measurements. *Renewable Energy*, *143*, 1643–1658. https://doi.org/10.1016/j.renene.2019.05.069.

Cao, T., Shen, Z., & Zhang, G. (2020). LSTM-Aided Reinforcement Learning for Energy Management in Microgrid with Energy Storage and EV Charging. *15th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN)*. https://doi.org/10.1109/MSN48538.2019.00017.

Chen, Y., Zhang, S., Zhang, W., Peng, J., & Cai, Y. (2019). Multifactor spatio-temporal correlation model based on a combination of convolutional neural network and long short-term memory neural network for wind speed forecasting. *Energy Conversion and Managemnet*, *185*, 783–799. https://doi.org/10.1016/j.enconman.2019.02.018.

Das, S. (2020). Short term forecasting of solar radiation and power output of 89.6kWp solar PV power plant. *Materials Today Proceedings*. https://doi.org/10.1016/j.matpr.2020.08.449.

Dong, N., Chang, J. F., Wu, A. G., & Gao, Z. K. (2020). A novel convolutional neural network framework based solar irradiance prediction method. *International Journal of Electrical Power & Energy Systems*, *114*. https://doi.org/10.1016/j.ijepes.2019.105411.

Dong, Z., Yang, D., Reindl, T., & Walsh, W. M. (2014). Satellite image analysis and a hybrid ESSS/ANN model to forecast solar irradiance in the tropics. *Energy Conversion and Management*, *79*, 66–73. https://doi.org/10.1016/j.enconman.2013.11.043.

Euskalmet (d). Agencia Vasca de meteorología. https://www.euskalmet.euskadi.eus/ (Accessed, February 2021).

Feng, C., & Zhang, J. (2020). SolarNet: A sky image-based deep convolutional neural network for intra-hour solar forecasting. *Solar Energy*, *204*, 71–78. https://doi.org/10.1016/j.solener.2020.03.083.

Gao, B., Huang, X., Shi, J., Tai, Y., & Zhang, J. (2020a). Hourly forecasting of solar irradiance based on CEEMDAN and multi-strategy CNN-LSTM neural networks. *Renewable Energy*, *162*, 1665–1683. https://doi.org/10.1016/j.renene.2020.09.141.

Gao, B., Huang, X., Shi, J., Tai, Y., & Zhang, J. (2020b). Hourly forecasting of solar irradiance based on CEEMDAN and multi-strategy CNN-LSTM neural networks. *Renewable Energy*, *162*, 1665–1683. https://doi.org/10.1016/j.renene.2020.09.141.

Glass, E., & Glass, V. (2020). Enabling supercapacitors to compete for ancillary services: An important step towards 100% renewable energy. *The Electricity Journal*, *33*. https://doi.org/10.1016/j.tej.2020.106763.

Gutierrez-Corea, F. V., Manso-Callejo, M. A., Moreno-Regidor, M. P., & Manrique-Sancho, M. T. (2016). Forecasting short-term solar irradiance based on artificial neural networks and data from neighboring meteorological stations. *Solar Energy*, *134*, 119–131. https://doi.org/10.1016/j.solener.2016.04.020.

Hyndman, R. J., & Fan, S. (2010). Density forecasting for long-term peak electricity demand. *IEEE Transactions on Power Systems*, *25*, 1142–1153. https://doi.org/10.1109/TPWRS.2009.2036017.

IEA (d). International Energy Agency. Global energy demand rose by 2.3% in 2018, its fastest pace in the last decade. https://www.iea.org/news/global-energy-demand-rose-by-23-in-2018-its-fastest-pace-in-the-last-decade (Accessed, February 2021).

IRENA (d). International Renewable Energy Agency. https://www.irena.org/ (Accessed, February 2021).

Ízgi, E., Öztopal, A., Yerli, B., Kaymak, M. K., & Şahin, A. D. (2012). Short–mid-term solar power prediction by using artificial neural networks. *Solar Energy*, *86*, 725–733. https://doi.org/10.1016/j.solener.2011.11.013.

Kalogirou, S. A. (2001). Artificial neural networks in renewable energy systems applications: a review. *Renewable and Sustainable Energy Reviews*, *5*, 373–401. https://doi.org/10.1016/S1364-0321(01)00006-5.

Koprinska, I., Wu, D., & Wang, Z. (2018). Convolutional Neural Networks for Energy Time Series forecasting. *International Joint Conference on Neural Networks (IJCNN)*. https://doi.org/10.1109/IJCNN.2018.8489399.

Kromer, P., Musílek, P., Pelikán, E., Krc, P., Jurus, P., & Eben, K. (2014). Support Vector Regression of multiple predictive models of downward short-wave radiation. *International Joint Conference on Neural Networks (IJCNN)*. https://doi.org/10.1109/IJCNN.2014.6889812.

Li, J., Geng, D., Zhang, P., Meng, X., Liang, Z., & Fan, G. (2019). Ultra-short term wind power forecasting based on LSTM neural network. *IEEE 3rd International Electrical and Energy Conference*. https://doi.org/10.1109/CIEEC47146.2019.CIEEC-2019625.

Lin, K. P., & Pai, P. F. (2016). Solar power output forecasting using evolutionary seasonal decomposition least-square support vector regression. *Journal of Cleaner Production*, *134*, 456–462. https://doi.org/10.1016/j.jclepro.2015.08.099.

Ludwing, S. A. (2019). Comparison of Time Series Approaches applied to Greenhouse Gas Analysis: ANFIS, RNN, and LSTM. *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. https://doi.org/10.1109/FUZZ-IEEE.2019.8859013.

Mathiesen, P., & Kleissl, J. (2011). Evaluation of numerical weather prediction for intra-day solar forecasting in the continental United States. *Solar Energy*, *85*, 967–977. https://doi.org/10.1016/j.solener.2011.02.013.

Mukhoty, B. P., Maurya, V., & Shukla, S. K. (2019). Sequence to sequence deep learning models for solar irradiation forecasting. *IEEE Milan PowerTech*. https://doi.org/10.1109/PTC.2019.8810645.

Ploysuwan, T. (2019). Deep CNN & LSTM network for appliances energy forecasting in residential houses using IoT sensors. *7th International Electrical Engineering Congress (IEECON)*. https://doi.org/10.1109/iEECON45304.2019.8938914.

Raza, M. Q., & Khosravi, A. (2015). A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings. *Renewable and Sustainable Energy Review*, *50*, 1352–1372. https://doi.org/10.1016/j.rser.2015.04.065.

Raza, M. Q., Nadarajah, M., & Ekanayake, C. (2016). On recent advances in PV output power forecast. *Solar Energy*, *136*, 125–144. https://doi.org/10.1016/j.solener.2016.06.073.

Rodríguez, F., Fleetwood, A., Galarza, A., & Fontán, L. (2018). Predicting solar energy generation through artificial neural networks using weather forecasts for microgrid control. *Renewable Energy*, *126*, 855–864. https://doi.org/10.1016/j.renene.2018.03.070.

Ryu, A., Ito, M., Ishii, H., & Hayashi, Y. (2019). Preliminary analysis of short-term solar irradiance forecasting by using Total-sky Imager and convolutional neural network. *IEEE PES GTD Grand International Conference and Exposition Asia*. https://doi.org/10.1109/GTDAsia.2019.8715984.

Shi, J., Lee, W. J., Liu, Y., Yang, Y., & Wang, P. (2011). Forecasting power output of photovoltaic system based on weather classification and support vector machine. *IEEE Industry Applications Society Annual Meeting*. https://doi.org/10.1109/IAS.2011.6074294.

Soman, S. S., Zareipour, H., Malik, O., & Mandal, P. (2010). A review of wind power and wind speed forecasting methods with different time horizons. *North America Power Symposium*. https://doi.org/10.1109/NAPS.2010.5619586.

Tiwari, S., Sabzehgar, R., & Rasoli, M. (2018). Short term solar irradiance forecast using numerical weather prediction (NWP) with gradient boost regression. *9th IEEE International Symposium on Power Electronics for Distributed Generation Systems (PEDG)*. https://doi.org/10.1109/PEDG.2018.8447751.

Wang, F., Xuan, Z., Zhen, Z., Li, Y., Li, K., Zhao, L., ... Catalão, J. P. S. (2020). A minutely solar irradiance forecasting method based on real-time sky image-irradiance mapping

model. *Energy Conversion and Management*, *220*. https://doi.org/10.1016/j.enconman.2020.113075.

Wang, F., Zhang, Z., Chai, H., Yu, Y., Lu, X., Wang, T., & Lin, Y. (2019). Deep learning based irradiance mapping model for solar PV power forecasting using sky image. *IEEE Industry Applications Society Annual Meeting*. https://doi.org/10.1109/IAS.2019.8912348.

Wang, H., Liu, Y., Zhou, B., Li, C., Cao, G., Voropai, N., & Barakhtenko, E. (2020). Taxonomy research of artificial intelligence for deterministic solar power forecasting. *Energy Conversion and Management*, *214*. https://doi.org/10.1016/j.enconman.2020.112909.

Wang, J., Guo, L., Zhang, C., Song, L., Duan, J., & Duan, L. (2020). Thermal power forecasting of solar power tower system by combining mechanism modeling and deep learning method. *Energy*, *208*. https://doi.org/10.1016/j.energy.2020.118403.

Wang, K., Qi, X., & Liu, H. (2019b). A comparison of day-ahead photovoltaic power forecasting models based on deep learning neural network. *Applied Energy*, *251*. https://doi.org/10.1016/j.apenergy.2019.113315.

Wang, K., Qi, X., & Liu, H. (2019c). Photovoltaic power forecasting based LSTM-Convolutional Network. *Energy*, *189*. https://doi.org/10.1016/j.energy.2019.116225.

Yan, K., Li, W., Ji, Z., Qi, M., & Du, Y. (2019). A hybrid LSTM neural network for energy consumption forecasting of individual households. *IEEE Access*, *7*, 157633–157642. https://doi.org/10.1109/ACCESS.2019.2949065.

Yi Hong, Y., Martinez, J. J. F., & Fajardo, A. C. (2020). Day-ahead solar irradiation forecasting utilizing gramian angular field and convolutional long short-term memory. *IEE Access*, *8*, 1841–18753. https://doi.org/10.1109/ACCESS.2020.2967900.

Zang, H., Cheng, L., Ding, T., Cheung, K. W., Liang, Z., Wei, Z., & Sun, G. (2018). Hybrid method for short-term photovoltaic power forecasting based on deep convolutional neural network. *IET Generation, Transmission & Distribution*, *12*, 4557–4567.

Zang, H., Cheng, L., Ding, T., Cheung, K. W., Wei, Z., & Sun, G. (2020). Day-ahead photovoltaic power forecasting approach based on deep convolutional neural networks and meta learning. *International Journal of Electrical Power & Energy Systems*, *118*. https://doi.org/10.1016/j.ijepes.2019.105790.

Zang, H., Liu, L., Sun, L., Cheng, L., Wei, Z., & Sun, G. (2020). Short-term global horizontal irradiance forecasting based on a hybrid CNN-LSTM model with spatiotemporal correlations. *Renewable Energy*, *160*, 26–41. https://doi.org/10.1016/j.renene.2020.05.150.

Zhoa, X., Jiang, N., Liu, J., Yu, D., & Chang, J. (2020). Short-term average wind speed and turbulent standard deviation forecasts based on one-dimensional convolutional neural network and the integrate method for probabilistic framework. *Energy Conversion and Management*, *203*. https://doi.org/10.1016/j.enconman.2019.112239.