Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2020) 28: 211 – 223
© TÜBİTAK
doi:10.3906/elk-1907-218

Time series forecasting on multivariate solar radiation data using deep learning (LSTM)

Murat Cihan SORKUN[1,2], **Özlem DURMAZ İNCEL**[2,*], **Christophe PAOLI**[2,3]

[1]Center for Computational Energy Research, Dutch Institute for Fundamental Energy Research, Eindhoven, Netherlands

[2]Department of Computer Engineering, Faculty of Engineering and Technology, Galatasaray University, İstanbul, Turkey

[3]University of Corsica Pasquale Paoli, Corte, France

Received: 29.07.2019 • **Accepted/Published Online:** 11.09.2019 • **Final Version:** 27.01.2020

Abstract: Energy management is an emerging problem nowadays and utilization of renewable energy sources is an efficient solution. Solar radiation is an important source for electricity generation. For effective utilization, it is important to know precisely the amount from different sources and at different horizons: minutes, hours, and days. Depending on the horizon, two main classes of methods can be used to forecast the solar radiation: statistical time series forecasting methods for short to midterm horizons and numerical weather prediction methods for medium- to long-term horizons. Although statistical time series forecasting methods are utilized in the literature, there are a limited number of studies that utilize deep artificial neural networks. In this study, we focus on statistical time series forecasting methods for short-term horizons (1 h). The aim of this study is to discover the effect of using multivariate data on solar radiation forecasting using a deep learning approach. In this context, we propose a multivariate forecast model that uses a combination of different meteorological variables, such as temperature, humidity, and nebulosity. In the proposed model, recurrent neural network (RNN) variation, namely a long short-term memory (LSTM) unit is used. With an experimental approach, the effect of each meteorological variable is investigated. By hyperparameter tuning, optimal parameters are found in order to construct the best models that fit the global solar radiation data. We compared the results with those of previous studies and we found that the multivariate approach performed better than the previous univariate models did. In further experiments, the effect of combining the most effective parameters was investigated and, as a result, we observed that temperature and nebulosity are the most effective parameters for predicting future solar radiance.

Key words: Deep learning, LSTM, solar radiation, time series

1. Introduction

The demand on energy is increasing day by day, but the energy resources are not sufficient to meet this demand. Nonrenewable energy resources, such as coal, oil, and natural gas, are also experiencing shortages and accordingly the trend towards renewable energy sources in the global sense has emerged. If utilized efficiently, solar energy has higher potential compared to the other resources. Technological developments and the reduction in infrastructure costs make it easier to make use of solar energy. Accordingly, it is becoming more and more important to precisely and efficiently forecast the availability of solar radiation.

*Correspondence: odincel@gsu.edu.tr

One disadvantage of solar energy-based systems is that the amount of solar radiation cannot be easily estimated since it depends on many variables. Time series forecasting can make it possible in short horizon prediction, such as a couple of hours. There are a number of models being used to forecast solar radiation with time series, such as the combination of autoregressive and dynamical system models [1], autoregressive integrated moving average (ARIMA) model [2], nonlinear autoregressive neural network [3], and multilayer perceptrons [4, 5].

One alternative method would be to use deep learning models, thanks to the disappearance of the hardware boundaries. In the literature, there are many deep learning models applied to time series data. These include deep belief networks [6–8], stacked auto encoders [8, 9], and long short-term memory (LSTM) units [9–11].

In the present study, we propose a multivariate forecast model that uses a combination of different meteorological variables such as temperature, humidity, and nebulosity. In the proposed model recurrent neural network (RNN) variation, namely a LSTM unit, is used. With an experimental approach, the effect of each meteorological variable is investigated. By hyperparameter tuning optimal parameters are found in order to construct the best models that fit the global solar radiation data. In further experiments, the effect of using the most effective parameters together is investigated. The results are compared with those of the univariate experiments. The main contributions and findings of the present study are listed as follows:

- We applied LSTM to multivariate meteorological time series data;
- We examined the effect of each individual meteorological parameter;
- We optimized the model in order the find the best hyperparameters of the structure.

The rest of the paper is organized as follows. Section 2 introduces the related studies that make use of time series forecasting for solar radiation data. The methodology, namely the deep learning methods, are introduced in Section 3. Section 4 contains the experiments using deep Learning methods for time series forecasting. Finally, Section 5 contains the conclusions drawn from the findings and includes projections for future studies.

2. Related work

In this section, we summarize the related studies that use time series forecasting methods for solar radiation data.

The magnitude of solar radiation can be affected by other meteorological variables. Many studies are conducted to reveal the relation between solar radiation and other meteorological variables. Most of them found that there is a strong dependency. Sfetsos and Coonick [12] compared univariate and multivariate approaches to solar radiation forecasting in their study. They used temperature, pressure, wind speed, and wind direction as additional parameters in their models. They reported that extra parameters improve the performance of adaptive neurofuzzy inference and ANN models.

In [13] the most relevant parameters for solar radiation prediction models are investigated. They identified temperature, altitude, and sunshine hours as the effective parameters. Chen and Li [14] used sunshine duration, temperature, humidity, and pressure as the attributes in their SVM model. They observed that sunshine duration and temperature significantly improve the model. In [15] Sun et al. examined the relationships between solar radiation and meteorological variables and they found a strong correlation between them. They also observed a higher correlation between solar radiation and sunshine duration than temperature.

More recent studies that use LSTM have been conducted during the past couple of years. In [16, 17] LSTM was used for solar radiation forecast. In another study [16], the dataset consists of 2.5 years with multiple variables, namely temperature, dew point, humidity, visibility, wind speed, and weather type. It was found that LSTM performs better than multilayered feedforward neural networks and linear regression. In another previous study [17], the dataset consists of only 4 days but it is high resolution (100 Hz). LSTM was compared with feedforward neural networks and support vector regression models and LSTM outperformed all other models. In [18] Abdel-Nasser and Mahmoud proposed 5 different photovoltaic power forecasting methods based on LSTM. Their dataset contains hourly data for a year collected from 2 cities in Egypt. They compared LSTM with feedforward neural networks, linear regression, and bagged regression trees and they found LSTM performs significantly better than the other models. However, different and smaller datasets were utilized in these studies. The effect of multivariate parameters was not investigated in these papers, unlike our study. Moreover, the effect of parameter tuning was not discussed either.

Finally, in [19–21] the same dataset is used as in our study. In [19] multivariate experiments are conducted for forecasting daily solar radiation using an ANN model. They observed that using multivariate data reduced the error rate between 0.5% and 1% (NRMSE). Lauret et al. [20] used a traditional ARMA model and lagged ANN structures to forecast hourly solar radiation. In our previous study [21] we utilized LSTM on the same dataset. However, in that study, only univariate experiments were conducted and, in the present study, we show that using multivariate data significantly improves the results.

3. Methodology: time series forecasting and deep learning

Time series are sequential data that are measured at certain intervals with respect to any process. The intervals used in the time series may be of different sizes, provided that they are equally divided. They are usually measured at intervals such as hourly, daily, monthly, and yearly. The annual population, the daily number of passengers on the metro, and the hourly exchange rate are examples of time series. In the time series, records must be ordered chronologically. Each record may contain information about one or more features. The term univariate is used for time series data containing single information, and multivariate is used for data containing more than one type of information.

In this context, time series forecasting can be defined as the prediction of future data using time series data of the past. Time series data are used for creating models by different methods. The process of creating models by formulating the data is called time series analysis. Forecasting is carried out through these models.

We can divide time series forecast models into two categories, linear and nonlinear. Linear models produce forecasts by taking the linear composition of past observations. Examples of linear modeling are regression, ARMA, and ARIMA. Nonlinear models are used when the data are multivariate or contain complex relationships in which linear models are insufficient to fit. Examples of nonlinear models are ANN, SVM, and decision trees. Another method used to forecast time series is naive forecasting. It is called naive because it uses only the last instance and ignores the historical data. This method uses the last observed value as the next prediction. Although it can achieve successful results over short horizon and low variance time series, it does not yield reliable results. Generally, it is used as a benchmark to compare other methods' performances.

In the following, we present the methodology that is followed in the present paper. First, we present the details of the dataset, then we present the LSTM model that is adopted, and finally we present the performance metric utilized in the experiments.

3.1. The data

In the present study, hourly global horizontal solar radiation data with extra meteorological parameters are used. Table 1 shows the explanations, units, and abbreviations of the parameters. The data cover 87,600 instances for the 10-year period from January 1998 to December 2007. Measurements of data provided by the French meteorological organization (Météo-France) were carried out at the meteorological station of Ajaccio (Corsica Island, France, 41°55′N, 8°44′E, 4 m above mean sea level) as demonstrated in Figure 1. The station is located between the Mediterranean Sea and the mountains. The sensors used in the station can work in the range of 0–90,000 J/m^2 and annual maintenance is done regularly. The location has a 'Mediterranean' climate, i.e. hot summers with abundant sunshine and mild, dry, clear winters.

Table 1. Explanation of meteorological parameters.

Parameter	Explanation	Measurement unit	Abbreviation
Temperature	Temperature under shelter	°C	TEMP
Pressure	Atmospheric pressure	Pa	PRES
Humidity	Relative humidity	%	HUM
Nebulosity	Total nebulosity	Octas	NEB
Wind Speed	Mean wind speed at 10 m	m/s	WS
Wind Direction	Wind direction at 10 m	360°	WD
Insolation	Sunshine duration (120 W m^2)	min	INS
Rain	The height of rain	mm	RAIN

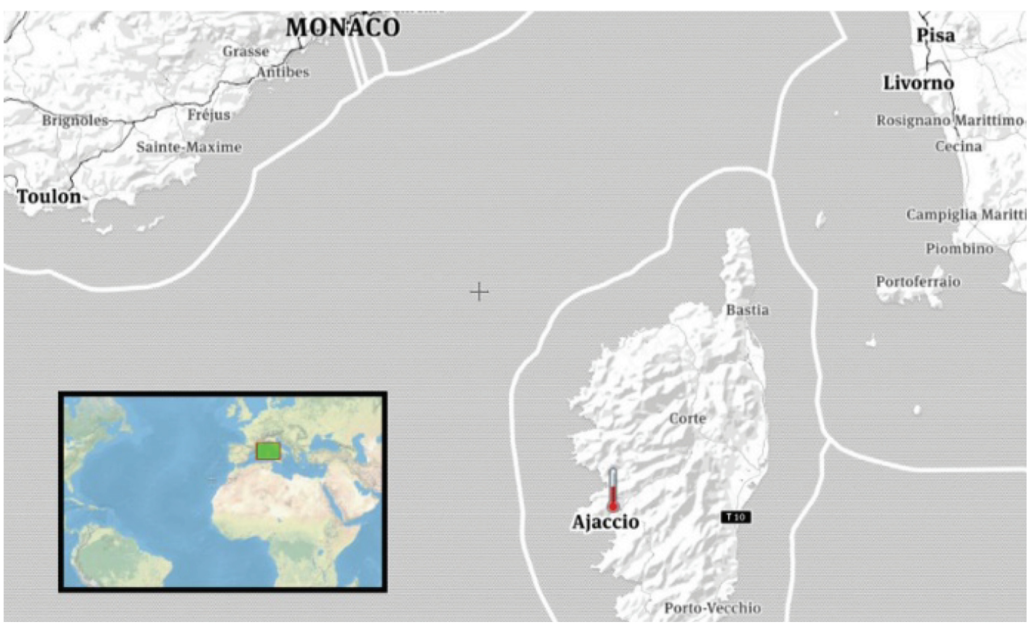


Figure 1. Meteorological station of Ajaccio.

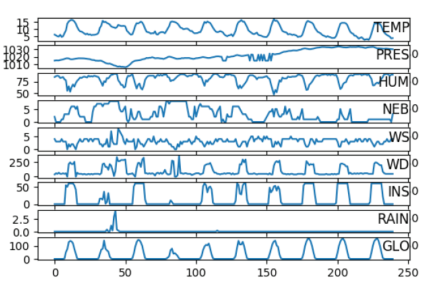
In Table 2 we present some statistical summaries about the parameters and in Table 3 we present the correlations between these parameters. According to the correlation matrix results, global solar radiation has a strong positive correlation with insolation; an intermediate positive correlation with temperature, wind speed, and wind direction; an intermediate negative correlation with humidity; and a weak correlation with pressure, nebulosity, and rain. Figures 2a–2c illustrate the pattern of each parameter for 10 days, 1 year, and 10 years, respectively. According to the patterns shown in Figure 2, all meteorological variables have daily seasonality except PRES, NEB, and RAIN. This seasonality has the same frequency with GLO. In Figure 2b TEMP, PRES, and GLO also show annual seasonality. As the data include annual seasonality, it would be reasonable to separate the test set to be at least one year or multiple.

Table 2. Analysis of meteorological parameters.

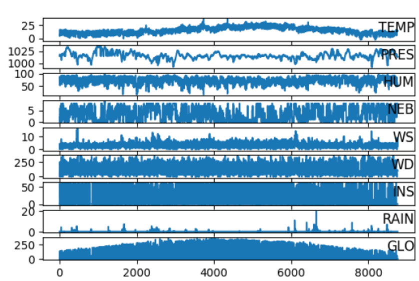
Parameter	Minimum	Maximum	Mean	Standard deviation
TEMP	–4.2	38.4	15.59	6.88
PRES	981.5	1039	1014.88	6.57
HUM	8	99	73.54	14.12
NEB	0	99	3.63	2.83
WS	0	21	3.43	1.77
WD	0	360	131.29	95.25
INS	0	60	19.1	26.08
RAIN	0	30	0.06	0.53
GLO	0	369	66.26	96.35

Table 3. Correlation matrix of meteorological parameters.

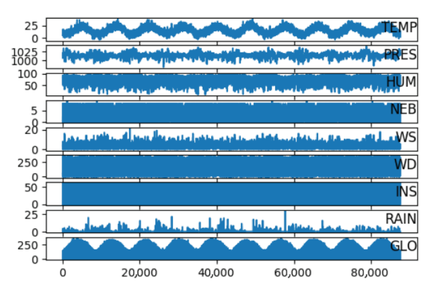
	TEMP	PRES	HUM	NEB	WS	WD	INS	RAIN	GLO
TEMP	1	–0.14	–0.45	–0.12	0.11	0.47	0.53	–0.04	0.58
PRES	–0.14	1	0.02	–0.3	–0.13	–0.15	0.09	–0.15	0.01
HUM	–0.45	0.02	1	0.13	–0.19	–0.47	–0.55	0.14	–0.52
NEB	–0.12	–0.3	0.13	1	–0.05	0.06	–0.27	0.17	–0.15
WS	0.11	–0.13	–0.19	–0.05	1	0.31	0.18	0.03	0.35
WD	0.47	–0.15	–0.47	0.06	0.31	1	0.58	0.01	0.64
INS	0.53	0.09	–0.55	–0.27	0.18	0.58	1	–0.09	0.85
RAIN	–0.04	–0.15	0.14	0.17	0.03	0.01	–0.09	1	–0.07
GLO	0.58	0.01	–0.52	–0.15	0.35	0.64	0.85	–0.07	1



(a) For 10 days



(b) For 1 year



(c) For 10 years

Figure 2. Patterns of meteorological parameters.

3.2. LSTM

Recurrent neural networks (RNNs) are a special kind of neural network developed to process sequential data. The first studies were conducted in the 1980s [22, 23]. In traditional structures, each sample is trained independently of each other, but this training is not a sufficient method for text, sound, image, or other data related to time. Independent training is not enough to preserve this knowledge because sequence information is also contained within sequence data. RNNs offer this probing solution by taking inputs sequentially.

Unlike feedforward neural networks, RNNs have feedback connections in the hidden layer units. With this feature, they can perform temporal processing and learn sequences. The hidden layer acts as a memory and can store sequential information. The RNNs' architecture can be transformed into an FNN structure by spreading over time. On this track, the RNNs can be trained with a back-propagation through time (BPTT) learning algorithm.

LSTM is a customized RNN model developed by Hochreiter and Schmidhuber [25]. LSTMs can keep track of long-term information through the gates they contain. In an LSTM unit there are basically three gates, which determine what information to store, namely the input gate, forget gate, and output gate. The input gate specifies which of the incoming input values will be stored in the next state. The forget gate determines which of the previous state information will no longer be stored. The output gate specifies which of the information in the new state will be sent as output. The gates that make up the LSTM unit are shown in Figure 3, where i, f, and o represent input gate, forget gate, and output gate, respectively. C represents the unit state and $\hat{C}$ represents the next candidate state.

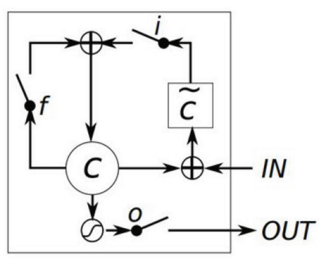


Figure 3. LSTM unit [24].

The equations used to calculate the next output and state values in the LSTM unit are as follows:

$$f_t = \sigma(W_f * [x(t), C(t-1), h(t-1)] + b_f) \tag{1}$$

$$i_t = \sigma(W_i * [x(t), C(t-1), h(t-1)] + b_i) \tag{2}$$

$$o_t = \sigma(W_o * [x(t), C(t), h(t-1)] + b_o) \tag{3}$$

$$C(t) = C(t-1) * f_t + \hat{C} * i_t, \tag{4}$$

where σ is the activation function, $x(t)$ is the input, $h(t-1)$ is the previous output, and W_{i}, W_{f}, W_{o} and b_{i}, b_{f}, b_{o} are the weights and biases of the input gate, forget gate, and output gate, respectively. We provide the details of the LSTM parameters used in the performance analysis in Sections 4.1 and 4.2, particularly in Figure 4.

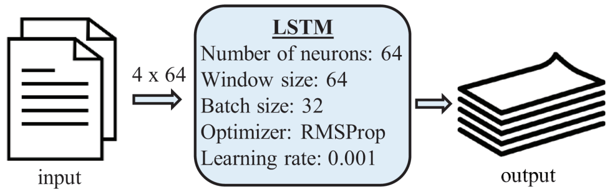


Figure 4. LSTM structure.

3.3. Performance metric

As the performance metric, we utilize the normalized root mean squared error (NRMSE). NRMSE is a normalized version of the root mean square error (RMSE). NRMSE can be calculated in several different ways. The most common of these is dividing the result obtained by RMSE by the average observed value. Calculation of NRMSE is given in Equation 5, where n is the number of instances, y is the actual value of instance, and e is the error of each forecast. NRMSE does not distinguish between negative and positive error and extreme errors are penalized by NRMSE. The NRMSE value should be kept as small as possible so that the estimates can be considered successful.

$$NRMSE = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n} {e_i}^2}}{\frac{1}{n}\sum_{i=1}^{n} y_i} \tag{5}$$

4. Experiments and results

The aim of the experiments presented in this section is to investigate the forecasting performance of LSTM for solar radiation using multivariate data. In the experimental process, first, the data are preprocessed and prepared for the training phase. Second, the effect of each meteorological variable is investigated. For this purpose 7 different models (for each parameter) are trained and their hyperparameters are optimized. Finally, the most effective parameters are selected and used to train the final model. The results obtained are compared with those of the univariate experiments, lagged neural networks, ARMA model, and naive method [20, 21].

In the preprocessing step, missing data are filled with the mean of the previous and next value except pressure (between 1 and 30 instances). The pressure data are filled with global mean (750 instances). After the missing values are filled, the data are normalized and values are scaled into [0,1] range. Finally, all zero values are removed and the data are split into a train set (80%) and a test set (20%).

All the experiments are conducted on Python using the TensorFlow framework and Keras neural network library.

4.1. Effect of meteorological variables and parameter tuning

The first set of experiments are conducted to investigate the effect of each meteorological variable. For this purpose, 7 different LSTM models are trained using each couple, for example, solar radiation and temperature. Only wind speed and wind direction data are used together. Each model is optimized through hyperparameter tuning. In order to optimize the models, the effect of the parameters through the experiments on the model performance was observed. The examined parameters are window size (previous sequenced values as number of inputs), number of neurons, and number of epochs. Each parameter is increased by doubling starting from 4 until 64 or there is no improvement in performance. We limited the maximum value of parameters because of the computational costs. The experiments were run 6 times for the same configuration in order to avoid local minimums. The best and mean results obtained from the results are shown in Tables 4–10. In all experiments, single layer LSTM is used with the following parameters: batch size is 32, optimizer is RMSProp, and learning rate is 0.001. Normalized root mean squared error (NRMSE) is used as an error metric.

Table 4. Hyperparameter tuning - effect of temperature.

Number of neurons	Number of epochs	Window size	Best (NRMSE)	Mean (NRMSE)
16	16	4	0.225124	0.227863
16	16	8	0.211028	0.214515
16	16	16	0.205737	0.209124
16	16	32	0.203789	0.209114
4	16	32	0.223622	0.224808
8	16	32	0.212922	0.215250
32	16	32	0.192241	0.195718
64	16	32	0.190752	0.191547
64	4	32	0.209618	0.213008
64	8	32	0.201242	0.201866
64	32	32	0.186055	0.188991
64	64	32	0.184857	0.185768
64	32	64	0.180151	0.182558

Table 5. Hyperparameter tuning - effect of pressure.

Number of neurons	Number of epochs	Window size	Best (NRMSE)	Mean (NRMSE)
16	16	4	0.244368	0.246037
16	16	8	0.232290	0.235710
16	16	16	0.225132	0.225886
16	16	32	0.213965	0.216785
4	16	32	0.231727	0.233618
8	16	32	0.222771	0.225194
32	16	32	0.212845	0.213338
64	16	32	0.209519	0.211350
64	4	32	0.217831	0.219996
64	8	32	0.211022	0.215780
64	32	32	0.206514	0.208251
64	64	32	0.209160	0.210872
64	32	64	0.204260	0.206099

According to the experimental results increasing the window size always increases the performance. This result shows that solar radiation is dependent on the past observations. The optimum number of neurons differs according to the model. In some experiments, 64 neurons provided the best performance, but in some cases there was a decrease in performance after 16 or 32. The observed difference can be interpreted such

Table 6. Hyperparameter tuning - effect of insolation.

Number of neurons	Number of epochs	Window size	Best (NRMSE)	Mean (NRMSE)
16	16	4	0.221680	0.222228
16	16	8	0.215752	0.220351
16	16	16	0.210573	0.214115
16	16	32	0.207011	0.209415
4	16	32	0.212666	0.217349
8	16	32	0.211012	0.211930
32	16	32	0.207440	0.209335
16	4	32	0.218466	0.222132
16	8	32	0.212833	0.214291
16	32	32	0.204696	0.206536
16	64	32	0.202771	0.203186
16	64	64	0.201077	0.201685

Table 7. Hyperparameter tuning - effect of humidity.

Number of neurons	Number of epochs	Window size	Best (NRMSE)	Mean (NRMSE)
16	16	4	0.240462	0.244287
16	16	8	0.231596	0.231979
16	16	16	0.221716	0.235797
16	16	32	0.211066	0.215217
4	16	32	0.229444	0.234037
8	16	32	0.224230	0.225946
32	16	32	0.207207	0.208249
64	16	32	0.205082	0.207706
64	4	32	0.216737	0.221553
64	8	32	0.208018	0.209430
64	32	32	0.201093	0.201513
64	64	32	0.204859	0.207934
64	32	64	0.200067	0.201434

that each meteorological parameter has a different correlation complexity with the future solar radiation level. This complexity affects the number of required neurons to represent the correlation. The best performance was obtained mostly at 32 epochs. This shows that models converge around 32 epochs and overfitting starts from 32 to 64.

4.2. Results with optimized parameters

After the effect of each parameter is found, a new set of experiments are conducted using the most effective parameters together. As shown in Table 11, TEMP and NEB are the most effective parameters and HUM and INS are the secondary effective parameters. First, the effect of TEMP and NEB parameters together is tested. Second, the secondary effective parameters are added individually to the most effective parameters. Finally,

Table 8. Hyperparameter tuning - effect of nebulosity.

Number of neurons	Number of epochs	Window size	Best (NRMSE)	Mean (NRMSE)
16	16	4	0.225443	0.227574
16	16	8	0.212332	0.213902
16	16	16	0.203267	0.206268
16	16	32	0.193451	0.195393
4	16	32	0.216532	0.218519
8	16	32	0.201421	0.205542
32	16	32	0.188164	0.189882
64	16	32	0.184978	0.187300
64	4	32	0.199013	0.201283
64	8	32	0.189622	0.192185
64	32	32	0.182640	0.187287
64	64	32	0.182544	0.183532
64	32	64	0.179581	0.179787

Table 9. Hyperparameter tuning - effect of rain.

Number of neurons	Number of epochs	Window size	Best (NRMSE)	Mean (NRMSE)
16	16	4	0.243712	0.246456
16	16	8	0.234794	0.235523
16	16	16	0.224182	0.227605
16	16	32	0.214572	0.220914
4	16	32	0.229604	0.233192
8	16	32	0.224452	0.225170
32	16	32	0.210155	0.213180
64	16	32	0.206708	0.209617
64	8	32	0.212522	0.214641
64	32	32	0.204403	0.206282
64	64	32	0.208317	0.209562
64	32	64	0.202923	0.203737

all parameters are used to train the model. The results of the experiments are shown in Table 12. Figure 4 illustrates the configured LSTM structure on multivariate experiments.

According to the results of the multivariate experiments,the parameters TEMP and NEB significantly improve the performance of the model. Other parameters slightly contribute to the performance. We obtain the best performance using the parameters GLO, TEMP, NEB, and HUM. As shown in Table 12, it is expected to improve the performance because there is a high correlation between GLO and TEMP and HUM. However, there was a low correlation between GLO and NEB but it was more effective than some other parameters with high correlation. The reason is the correlations show only the relationship on the instantaneous values and do not take into account the information from the past observations. If the past observations of the GLO already contain information about the new parameter, the effect will be low, even if the correlation is high. Similarly, low-correlated parameters may be more related to future information or they may contain new information.

Table 10. Hyperparameter tuning - effect of wind speed and direction.

Number of neurons	Number of epochs	Window size	Best (NRMSE)	Mean (NRMSE)
16	16	4	0.244531	0.249973
16	16	8	0.234594	0.237008
16	16	16	0.225131	0.227632
16	16	32	0.215414	0.221502
4	16	32	0.228492	0.232483
8	16	32	0.227567	0.227824
32	16	32	0.212512	0.216422
64	16	32	0.215104	0.209526
32	4	32	0.225321	0.227798
32	8	32	0.219678	0.225127
32	32	32	0.207562	0.211713
32	64	32	0.207213	0.208513
32	32	64	0.205194	0.206534

Table 11. Effect of each meteorological parameter.

Input parameters	Best (NRMSE)	Mean (NRMSE)
GLO, TEMP	0.180151	0.182558
GLO, PRES	0.204260	0.206099
GLO, HUM	0.200067	0.201434
GLO, NEB	0.179581	0.179787
GLO, WS, WD	0.205194	0.206534
GLO, INS	0.201077	0.201685
GLO, RAIN	0.202923	0.203737

Table 12. Effect of combined meteorological parameters.

Input parameters	Best (NRMSE)	Mean (NRMSE)
GLO,TEMP, NEB	0.160376	0.162538
GLO,TEMP, NEB, HUM	0.159261	0.161425
GLO,TEMP, NEB, INS	0.162142	0.163072
GLO,TEMP, NEB, HUM, INS	0.160638	0.163214
ALL	0.160397	0.160789

These reasons can also cause a high impact. Although the correlation matrix provides a general view, it does not provide sufficient and precise information. As a result, the contribution of parameters can be interpreted as temperature (TEMP) and nebulosity (NEB) contain more additional information correlated with the future solar radiance (GLO). Table 13 shows the comparison of the multivariate model with previous experiments [20, 21].

5. Conclusion

In this study, we applied LSTM models for time series forecasting on multivariate solar radiation data for 1-h horizon.

Table 13. Comparison of the multivariate model with the univariate model and the naive method.

Model	Result (NRMSE)
Naive method	0.37
ARMA [20]	0.2501
Neural networks [20]	0.2310
LSTM (univariate) [21]	0.213652
LSTM (multivariate)	0.159261

In the experiments, the effect of additional meteorological parameters is investigated. According to the results of multivariate experiments, the parameters temperature and nebulosity significantly improve the performance of the model. Other parameters contribute very slightly. We obtain the best performance using the parameters global solar radiation, temperature, nebulosity, and humidity.

Because of the longevity of the experiments, in hyperparameter tuning, the number of neurons, the number of epochs, and the window size are optimized, and the optimization of other parameters such as learning rate, batch size, and activation function are considered as future work.

The results show that LSTM models can be suitable and competitive for 1-h horizon time series forecasting for multivariate solar radiation data. The experiments proved that solar radiation is strongly dependent on other meteorological parameters.

This study can be extended with possible future work:

- Perform training with one step ahead cross-validation;
- Investigation to the effect of different activation functions and optimizers;
- Optimization of parameters (learning rate and batch size);
- Testing the proposed model with data from different locations;
- Construction of hybrid architectures using multivariate data;
- Implementation of hidden Markov models for comparison.

Acknowledgment

Thanks to the University of Corsica, UMR CNRS 6134 for providing the data used in this study.

References

[1] Huang J, Korolkiewicz M, Agrawal M, Boland J. Forecasting solar radiation on an hourly time scale using a coupled autoregressive and dynamical system (CARDS) model. Solar Energy 2013; 87: 136-149.

[2] Yang D, Jirutitijaroen P, Walsh WM. Hourly solar irradiance time series forecasting using cloud cover index. Solar Energy 2012; 86 (12): 3531-3543.

[3] Benmouiza K, Cheknane A. Forecasting hourly global solar radiation using hybrid k-means and nonlinear autoregressive neural network models. Energy Conversion and Management 2013; 75: 561-569.

[4] Mihalakakou G, Santamouris M, Asimakopoulos DN. The total solar radiation time series simulation in Athens, using neural networks. Theoretical and Applied Climatology 2000; 66 (3-4): 185-197.

[5] Paoli C, Voyant C, Muselli M, Nivet ML. Forecasting of preprocessed daily solar radiation time series using neural networks. Solar Energy 2010; 84 (12): 2146-2160.

[6] Kuremoto T, Kimura S, Kobayashi K, Obayashi M. Time series forecasting using a deep belief network with restricted Boltzmann machines. Neurocomputing 2014; 137: 47-56.

[7] Qiu X, Zhang L, Ren Y, Suganthan PN, Amaratunga G. Ensemble deep learning for regression and time series forecasting. In: IEEE 2014 Computational Intelligence in Ensemble Learning (CIEL) Symposium; Orlando, FL, USA; 2014. pp. 1-6.

[8] Lv Y, Duan Y, Kang W, Li Z, Wang FY. Traffic flow prediction with big data: a deep learning approach. IEEE Transactions on Intelligent Transportation Systems 2015; 16 (2): 865-873.

[9] Bao W, Yue J, Rao Y. A deep learning framework for financial time series using stacked autoencoders and long-short term memory. PloS one 2017; 12 (7): 1-24.

[10] Fischer T, Krauss C. Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research 2018; 270 (2): 654-669.

[11] Hsu D. Time series forecasting based on augmented long short-term memory. arXiv preprint 2017; arXiv:1707.00666: 1-14.

[12] Sfetsos A, Coonick AH. Univariate and multivariate forecasting of hourly solar radiation with artificial intelligence techniques. Solar Energy 2000; 68 (2): 169-178.

[13] Yadav AK, Malik H, Chandel SS. Selection of most relevant input parameters using WEKA for artificial neural network based solar radiation prediction models. Renewable and Sustainable Energy Reviews 2014; 31: 509-519.

[14] Chen JL, Li GS. Evaluation of support vector machine for estimation of solar radiation from measured meteorological variables. Theoretical and Applied Climatology 2014; 115 (3-4): 627-638.

[15] Sun H, Zhao N, Zeng X, Yan D. Study of solar radiation prediction and modeling of relationships between solar radiation and meteorological variables. Energy Conversion and Management 2015; 105: 880-890.

[16] Qing X, Niu Y. Hourly day-ahead solar irradiance prediction using weather forecasts by LSTM. Energy 2018; 148: 461-468.

[17] Alzahrani A, Shamsi P, Dagli C, Ferdowsi M. Solar irradiance forecasting using deep neural networks. Procedia Computer Science 2017; 114: 304-313.

[18] Abdel-Nasser M, Mahmoud K. Accurate photovoltaic power forecasting models using deep LSTM-RNN. Neural Computing and Applications 2019; 31 (7): 2727-2740.

[19] Voyant C, Muselli M, Paoli C, Nivet ML. Optimization of an artificial neural network dedicated to the multivariate forecasting of daily global radiation. Energy 2011; 36 (1): 348-359.

[20] Lauret P, Rodler A, Muselli M, David M, Diagne H et al. A Bayesian model committee approach to forecasting global solar radiation. arXiv preprint 2012; arXiv:1203.5446: 1-6.

[21] Sorkun MC, Paoli C, Incel ÖD. Time series forecasting on solar irradiation using deep learning. In: IEEE Electrical and Electronics Engineering Conference (ELECO); Bursa, Turkey; 2017. pp. 151-155.

[22] Hopfield JJ. Neurons with graded response have collective computational properties like those of two-state neurons. In: Proceedings of the National Academy of Sciences 1984; 81 (10): 3088-3092.

[23] Rummelhart DE, Hinton GE, Williams RJ. Learning Internal Representations by Error Propagation. La Jolla, CA, USA: Institute for Cognitive Science, University of California, San Diego, 1985.

[24] Chung J, Gulcehre C, Cho K, Bengio Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint 2014; arXiv:1412.3555: 1-9.

[25] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Computation 1997; 9 (8): 1735-1780.