

Hourly Prediction of Irradiance and Temperature Using Recurrent Neural Networks and Gaussian Process Models

Predicción horaria de irradiancia y temperatura usando redes recurrentes y modelos gaussianos

Mónica Moreno Revelo ¹, Juan Gómez Mendoza ², and Javier Revelo Fuelagán ³

Fecha de Recepción: 27 de junio de 2025

Fecha de Aceptación: 26 de septiembre de 2025

Cómo citar: MY. Moreno-Revelo, J. Gómez-Mendoza, and J. Revelo-Fuelagán "Hourly Prediction of Irradiance and Temperature Using Recurrent Neural Networks and Gaussian Process Models," *Tecnura*, 29(85), 89-103. <https://doi.org/10.14483/22487638.23836>

Abstract


This research applies artificial intelligence techniques to predict physical variables such as irradiance and temperature, addressing the challenge of time series nonlinearity. The main objective is to compare the predictive performance of LSTM, GRU, and SGPR models across three datasets (Jena, Solcast, and IDEAM), evaluating three sliding-window configurations and an hourly prediction scheme, where a separate model is trained for each hour instead of relying on a single general model. Results, assessed using MAE, RMSE, and R^2 , show that in the Jena dataset, for example, the SGPR model under the hourly approach achieves average values of 0.53°C, 0.74°C, and 0.99, respectively. Overall, the findings suggest that employing multiple Gaussian process-based models trained with hour-specific information yields superior performance compared to using a single model.


Keywords: Hourly models, LSTM, GRU, Meteorological time series, Hourly prediction.

Resumen

Esta investigación aplica técnicas de inteligencia artificial para predecir variables físicas como irradiancia y temperatura, enfrentando el reto de la no linealidad en series de tiempo. El objetivo principal es comparar la eficiencia de modelos LSTM, GRU y SGPR en tres bases de datos (Jena, Solcast e IDEAM), evaluando tres configuraciones de ventanas deslizantes y un esquema de predicción horaria, en el cual se entrena un modelo independiente para cada hora en lugar de un único modelo general. Los resultados, medidos mediante MAE, RMSE y R^2 , muestran que, en la base de datos Jena, por ejemplo, el modelo SGPR bajo el enfoque horario alcanza valores promedio de 0.53°C, 0.74°C, y 0.99, respectivamente. En general, los hallazgos sugieren que emplear múltiples modelos basados en procesos gaussianos entrenados con información específica por hora, produce un rendimiento superior comparado con el uso de un solo modelo.

¹Magíster en ingeniería. Profesora del Departamento de Ingeniería Civil de la Universidad Mariana – Pasto, Nariño, Colombia.  Email: monmoreno125@umariana.edu.co

²PhD. en ingeniería. Profesor Universidad Nacional de Colombia  Email: jbgomez@unal.edu.co

³PhD. en ingeniería. Profesor Universidad de Nariño  Email: javierrevelof@udenar.edu.co

0.74°C y 0.99, respectivamente. En conjunto, los hallazgos sugieren que emplear múltiples modelos basados en procesos gaussianos con información específica por hora ofrece un desempeño superior al uso de un único modelo.

Palabras clave: Modelos horarios, LSTM, SGPR, series de tiempo meteorológicas, predicción horaria

Introduction

The prediction of physical variables such as irradiance and temperature plays a crucial role in areas like renewable energy utilization and energy consumption optimization in buildings, contributing to the reduction of fossil fuel dependency (1). This represents a significant contribution, considering that fossil fuels account for nearly 80 % of global energy production (2). This figure is particularly concerning because each kilowatt-hour (kWh) of energy generated from fossil fuels emits approximately 0.136 kg of carbon dioxide (CO₂), according to the Intergovernmental Panel on Climate Change (IPCC) (3). CO₂ emissions substantially contribute to environmental pollution by trapping heat in the atmosphere, thereby intensifying global warming and climate change (4).

Artificial Intelligence (AI) is increasingly playing a fundamental role across various fields. In particular, AI-driven time series prediction has been explored through three main approaches: physical models, data-driven models, and hybrid methods. (5–7). The survey presented in (8) provides a comprehensive overview of these techniques, highlighting their respective advantages and limitations.

Physical models, such as Numerical Weather Prediction (NWP), rely on physics-driven equations and can provide coverage for nearly all geographical regions. However, they often face challenges with short-term forecasting (9, 10). An example of such methods is presented in (11), where six equations are combined with Solcast to estimate irradiance across various cities in Morocco. Validation against data from local stations revealed that Solcast tends to overestimate irradiance in certain regions.

Data-driven methods (e.g., LSTM, GRU, GP) are effective at extracting patterns from historical datasets but require high-quality, relevant data. Unlike physical models, these approaches do not depend on expert climate knowledge, although they also have certain limitations (12,13). For instance, the authors in (14) propose a Long Short-Term Memory (LSTM) architecture and analyze the impact of various parameters, concluding that a configuration without output activation (NOA) achieves superior performance for energy-related applications. Similarly, in (15), a LSTM is employed to forecast irradiance using IDEAM (Instituto de Hidrología y Estudios Ambientales) data from Nariño region, demonstrating improved performance over other methods despite challenges related to significant missing data.

Hybrid methods combine different techniques to leverage the strengths of each resource but typically demand higher computational resources. For example, the researches in (16) utilize a Convolutional Neural Network (CNN) to extract local features from time series data, which are then processed by an LSTM network to predict the power output of a photovoltaic plant. Similarly, the authors in (17) apply this approach to predict meteorological variables using the Hawaii and Alice Springs datasets. Their work also incorporates a transformer-based architecture in the initial stage to handle missing data, followed by a prediction stage.

Given the limitations of existing methods in capturing essential temporal features and incorporating relevant input information, a novel methodology for predicting irradiance and temperature is proposed in this study. The approach involves comparing three models: Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Sparse Gaussian Process Regressor (SGPR). The innovation lies in utilizing direct models to predict the target variable for each hour individually, rather than relying on a single iterative model to forecast the variable across all time points. The results demonstrate that the direct models with SGPR substantially improve performance, achieving a Mean Absolute Error (MAE) of 0.53 °C and a Root Mean Square Error (RMSE) of 0.74° C on the Jena dataset.

This paper is organized as follows: Section 2 describes the dataset, methods, performance metrics, and experiments conducted. Section 3 presents the results obtained using the proposed methodology. Finally, Section 4 outlines the main conclusions of the study.

The methodology proposed is illustrated in Figure 1. First, a pre-processing step is applied to the available data, and the architectures of the three predictors are established. Next, the window configuration is determined using a correlation analysis. Finally, the models are evaluated, and 1-day-ahead predictions are made on the time series.

Methodology

Datasets

- The Jena dataset is a weather time series collected at the Weather Station of the Max Planck Institute. It includes 14 meteorological variables, such as temperature, humidity, and wind speed, recorded at 10-minute intervals. The dataset spans from January 2009 to December 2016.
- Solcast operates a prediction system that integrates near real-time satellite imagery from 11 weather satellites with data from 7 numerical weather prediction (NWP) models (18). The system provides 19 meteorological variables, including global horizontal irradiance

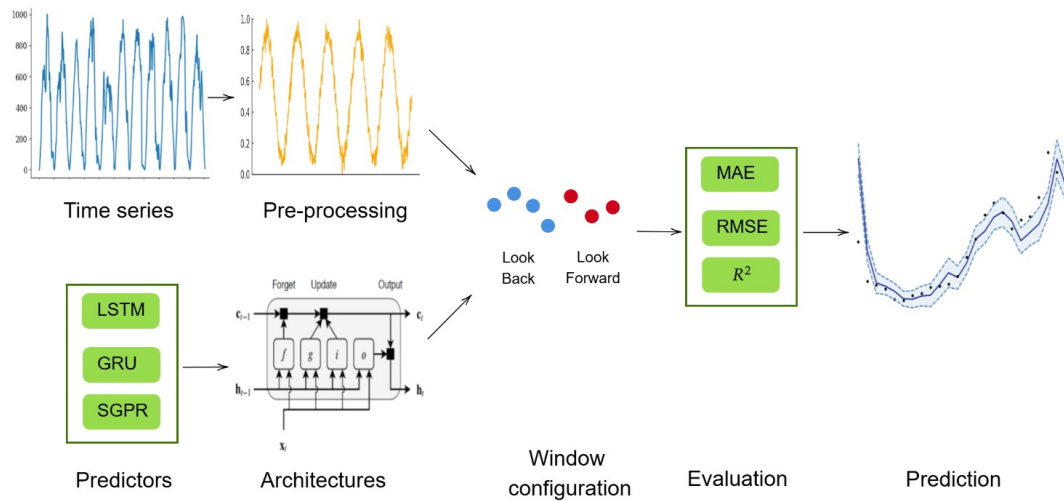


Figure 1. Methodology applied for irradiance and temperature prediction

Source: Authors.

Table 1. Irradiance and temperature datasets information

Dataset	Geographical coverage	Temporal coverage	Sampling time (minutes)	Variable
Jena	Jena-Germany	January 2009 to October 2015	10	Temperature (°C)
Solcast	Viento Libre-Nariño-Colombia	January 2006 to April 2024	5	Irradiance (W/m ²)
IDEAM	Viento Libre-Nariño-Colombia	November 2016 to March 2023	60	

Source: Authors.

(GHI), wind speed, and cloudiness, recorded at 5-minute intervals. The dataset spans from January 2006 onward, offering global spatial coverage, excluding the North Pole.

- IDEAM plays a crucial role in monitoring weather patterns to manage water resources and conduct environmental assessments in Colombia (15). The institute operates numerous weather stations distributed across the country, recording meteorological variables such as irradiance and precipitation, recorded at 60-minute intervals. However, the temporal coverage varies by station, and some periods contain significant gaps in the data.

In our experiments we use the features of the datasets described in Table 1.

Before applying the proposed methodology, a pre-processing step is performed on the available data:

- Time adjustment is applied to Solcast data to align it with the Colombian time zone.
- Removal of nighttime data is applied to the Solcast and IDEAM datasets, as irradiance is zero during this period.
- Repeated values are removed from Jena, as there are days with more than 145 recorded samples.
- Normalization is applied to the Solcast, IDEAM, and Jena datasets, scaling the values to a range between 0 and 1. Equation (1) describes the MinMax Scaler method used.

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (1)$$

where x' is the normalized data, x the original data, x_{\min} is the min value of the data, and x_{\max} is the max value of the data.

- Resampling to a 1-hour frequency is applied to the Solcast and Jena datasets, following the recommendation given in (14).

Predictors

The following three methods are employed to make predictions using the available datasets:

- Long Short-Term Memory (LSTM): LSTM is an enhanced version of Recurrent Neural Network (RNN) designed to retain both short-term and long-term dependencies (8). An LSTM unit consists of cells, each equipped with a forget gate (Equation (2)), an update gate (Equation (3)), and an output gate (Equation (4)).

$$f_t = \sigma(w_f[a_{t-1}, x_t] + b_f) \quad (2)$$

$$u_t = \sigma(w_u[a_{t-1}, x_t] + b_u) \quad (3)$$

$$o_t = \sigma(w_o[a_{t-1}, x_t] + b_o) \quad (4)$$

where x are the inputs, σ is the activation function, w are the weights, and b are the bias.

- Gated Recurrent Unit (GRU): GRU is another variant of RNN designed to address issues related to vanishing and exploding gradients. Compared to LSTM, GRU is simpler to

implement and compute (19). It incorporates two gates: an update gate (Equation (5)) and a reset gate (Equation (6)).

$$z_t = \sigma(w_z x_t + v_z h_{t-1} + b_z) \quad (5)$$

$$r_t = \sigma(w_r x_t + v_r h_{t-1} + b_r) \quad (6)$$

where x are the inputs, h_t are the hidden states, σ is the activation function, w and v are the weights, and b are the bias.

- Gaussian Process regressor (GPR): GPR is a regressor based on Gaussian process which is a collection of random variables defined by its mean (Equation (7)) and its kernel (Equation (8)) (20).

$$m(x) = \mathbb{E}[f(x)] \quad (7)$$

$$k(x, x') = \mathbb{E}[(f(x) - m(x))(f(x') - m(x')))] \quad (8)$$

where x and x' are the observed and the predicted values, respectively.

There are also variants of this method, such as the Sparse Gaussian Process Regressor (SGPR), which, rather than utilizing all the training samples, only uses a subset called inducing points.

The architectures of the models described in Table 2 are based on the research conducted by (21) and (22), without structural modifications. However, the parameters are tuned through experimental analysis, where each parameter is varied until the best results are obtained. Accordingly, Table 2 also lists the parameters used—these correspond to the configurations that achieve the highest accuracy. Likewise, training is performed using a 3-fold expanding window time-series cross-validation scheme.

Prediction over sliding temporal windows

Considering that many applications require predictions at multiple future time points, this research focuses on predicting irradiance or temperature values for 1 day ahead (short-term prediction). According to the literature, two primary approaches can be used for this type of prediction: direct and iterative methods.

- Direct methods involve making a prediction for a specific time without relying on previous predictions. As a result, a separate model must be trained for each specific time point.
- Iterative methods use previous predictions as inputs for subsequent predictions, requiring only a single model to make predictions for any time point.

Table 2. Architectures used for time series prediction

LSTM and GRU	
Inputs	Adaptive (irradiance or temperature)
Outputs	1(irradiance or temperature)
Number of layers	3 (input, hidden with 20 neurons, output)
Dropout	0.1 after the hidden layer
Activation functions	tanh in the hidden layer and linear in the output layer
Epochs	20
Learning algorithm	Adam (learning rate=0.07)
SGPR	
Inputs	Adaptive (irradiance or temperature)
Outputs	1(irradiance or temperature)
Kernel	RBF(lengthscales=1.0) + SquaredExponential
Number of inducing points	100
Iterations	1000
Framework	GPflow 2.6 (TensorFlow backend)
Optimizer	L-BFGS-B

Source: Authors.

Based on these approaches, various configurations of sliding temporal windows are evaluated, taking into account two key parameters: look-back, which describes the number of previous time steps used to predict future values; and look-forward, which specifies the number of future time steps to be predicted (7). Then, the following test **scenarios** are conducted:

- A **single** preceding sample predicts the **next** sample using an **iterative** model.
- **Multiple** preceding samples predict the **next** sample using an **iterative** model.
- **Multiple** preceding samples predict **multiple** future samples using an **iterative** model.
- **Multiple** preceding samples predict the **next** sample using **direct** models for each future time point.

In the third scenario, the look-back and look-forward parameters are configured based on recommendations given in (15). For the second and fourth scenarios, the look-back parameter is determined through a correlation index (see equation (9)).

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (9)$$

where x_i and y_i are the observed values of variables x and y , respectively, \bar{x} and \bar{y} are the mean values of x and y , n is the total number of observations, and \sum denotes the summation over all observations.

Performance measurements

Table 3 presents the metrics used to evaluate the performance of the proposed methodology. In the equations I_m is the measured value and I_p is the predicted value.

Table 3. Performance metrics used to evaluate the predictive accuracy of the models.

Metric	Abbreviation	Equation	Range
Mean Absolute Error	MAE	$\frac{1}{N} \sum_{i=1}^N I_m - I_p $	$[0, \infty]$
Root Mean Square Error	RMSE	$\sqrt{\frac{1}{N} \sum_{i=1}^N (I_m - I_p)^2}$	$[0, \infty]$
Determination Coefficient	R^2	$1 - \frac{\sum_{i=1}^N (I_m - I_p)^2}{\sum_{i=1}^N (I_m - \bar{I}_m)^2}$	$[0, 1]$

Source: Authors.

Results

This section presents the results obtained by applying the predictors described in Section 2.2 to the datasets detailed in Section 2.1. First, the results obtained using the “iterative” model are discussed. Next, the outcomes from the direct models are presented. Finally, some graphical examples are provided.

As mentioned in the *Methodology – Predictors* section, only the parameters of the architectures were modified to obtain the best results. For instance, in the LSTM model, it was observed that the learning rate was inversely related to prediction accuracy—lower learning rates tended to improve accuracy, while higher ones increased computational cost.

In the case of the SGPR model, adjustments were made to the kernel configuration and the initialization of the inducing points. Although the latter had little influence on accuracy, modifying the kernel significantly affected performance. Specifically, replacing the RBF kernel with alternatives such as Matern or Rational Quadratic led to a noticeable decrease in accuracy.

Iterative method

The LSTM, GRU, and SGPR architectures were utilized to make predictions in the first three scenarios outlined in Section 2.3. The corresponding results are summarized in Table 4.

Table 4. Prediction measures. MAE and RMSE values are expressed in °C for temperature (Jena), and in W/m² for irradiance (Solcast and IDEAM).

Test	Predictor	Dataset	LB/ LF	Training			Testing		
				MAE	RMSE	R ²	MAE	RMSE	R ²
1	LSTM	Jena	1/1	0.90 ± 0.11	1.19 ± 0.07	0.98 ± 0.01	0.91 ± 0.14	1.20 ± 0.14	0.98 ± 0.01
2	GRU	Jena	1/1	0.88 ± 0.21	1.17 ± 0.18	0.98 ± 0.01	0.89 ± 0.22	1.18 ± 0.21	0.98 ± 0.01
3	SGPR	Jena	1/1	0.72 ± 0.01	1.02 ± 0.01	0.98 ± 0.01	0.72 ± 0.03	1.02 ± 0.04	0.98 ± 0.01
4	LSTM	Jena	4/1	0.76 ± 0.12	0.99 ± 0.08	0.99 ± 0.01	0.74 ± 0.11	0.98 ± 0.09	0.99 ± 0.01
5	LSTM	Jena	240/24	1.99 ± 0.02	2.62 ± 0.04	0.91 ± 0.01	2.05 ± 0.10	2.72 ± 0.14	0.88 ± 0.01
6	LSTM	Solcast	8/1	76.33 ± 2.99	107.27 ± 2.16	0.87 ± 0.01	77.21 ± 3.02	107.55 ± 2.32	0.87 ± 0.01
7	LSTM	IDEAM	8/1	105.29 ± 4.86	146.13 ± 6.57	0.74 ± 0.02	114.07 ± 7.01	159.98 ± 10.90	0.68 ± 0.04

Source: Authors.

According to Table 4, in the first scenario (Tests 1, 2, and 3), the SGPR model delivered the best results. While the second (Tests 4, 6, and 7) and third scenarios (Test 5) were initially intended to be developed using the SGPR model, its memory requirements made this impractical. Consequently, the LSTM model was employed to evaluate the configurations of the sliding windows.

The results indicate that the worst performance occurred in the third scenario, where multiple preceding samples predict multiple future samples). Conversely, the best performance was observed in the second scenario, where multiple preceding samples predict the next sample. Consequently, the SGPR model achieved the highest performance overall, particularly when using a look-back period greater than one and a look-forward equal to one.

Direct method

Considering the results obtained from the iterative model, a direct method using the SGPR model and the Jena dataset was proposed for the fourth scenario described in Section 2.3, where memory requirements were not a limiting factor. The results are summarized in Table 5.

Table 5. Prediction measures-hourly methodology. MAE and RMSE values are expressed in °C.

Test	Hour	Range Time	LB/ LF	Training			Testing		
				MAE	RMSE	R ²	MAE	RMSE	R ²
8A	00	23-0	21/1	0.43 ± 0.01	0.58 ± 0.01	0.99 ± 0.01	0.45 ± 0.02	0.60 ± 0.04	0.99 ± 0.01
8B	04	03-04	21/1	0.36 ± 0.01	0.51 ± 0.01	0.99 ± 0.01	0.38 ± 0.01	0.52 ± 0.01	0.99 ± 0.01
8C	08	07-08	21/1	0.69 ± 0.01	0.93 ± 0.01	0.99 ± 0.01	0.72 ± 0.02	0.94 ± 0.02	0.98 ± 0.01
8D	12	11-12	21/1	0.62 ± 0.02	0.86 ± 0.03	0.99 ± 0.01	0.66 ± 0.01	0.92 ± 0.04	0.98 ± 0.01
8E	16	15-16	21/1	0.58 ± 0.01	0.90 ± 0.02	0.99 ± 0.01	0.57 ± 0.03	0.86 ± 0.10	0.99 ± 0.01
8F	20	19-20	21/1	0.45 ± 0.02	0.63 ± 0.03	0.99 ± 0.01	0.49 ± 0.03	0.68 ± 0.05	0.99 ± 0.01
Mean				0.52 ± 0.01	0.73 ± 0.01	0.99 ± 0.01	0.53 ± 0.02	0.74 ± 0.04	0.99 ± 0.01
(14)				0.53	0.77		0.52	0.75	

Source: Authors.

In this second phase, 24 models were developed to predict temperature for each hour of the day, from 00:00 to 23:00. In all cases, the look-back period was set to 21, but the selected samples varied depending on the hour being predicted. For instance, in the model labeled 8I, which predicts the temperature at 08:00, the samples from 07:00 to 08:00 across the previous ten days were utilized.

The results for specific hours are presented in Table 5, while the overall mean performance is calculated across all 24 models. Notably, the direct modeling approach for each hour significantly improved the predictions, achieving results comparable to those reported in the literature. Furthermore, the memory requirements were reduced, enabling the application of SGPR with a look-back period greater than one.

Graphical examples of the results

Finally, the predictions for a single day are presented graphically in Figures 2, 3, and 4, corresponding to Jena, Solcast, and IDEAM, respectively.

Figure 2 presents the temperature prediction results for December 24, 2012, spanning from 00:00 to 23:00 hours. The predictions were generated using the direct models. As shown, the predicted temperature curve closely follows the observed values, effectively capturing the diurnal variations in temperature. The largest discrepancies are primarily observed during the evening hours, when the temperature values are higher. The blue shaded region represents the 95 % confidence interval, within which the majority of the observed values are contained. This indicates that the model provides a reasonably accurate estimation of prediction uncertainty for the particular example provided.

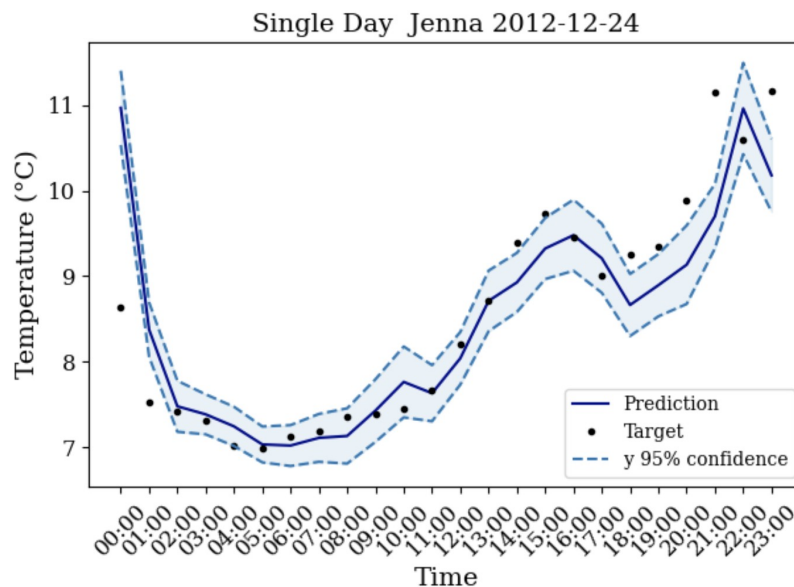


Figure 2. Predictions in the Jena dataset using the SGPR model applied on an hourly basis.

Source: Authors.

Figures 3 and 4 show the irradiance prediction results for August 10, 2015, covering the period from 06:00 to 18:00. The iterative model was utilized, employing the configurations from Test 6 and Test 7 for the Solcast and IDEAM datasets, respectively. Specifically, an LSTM model with a configuration of 8 previous samples to predict 1 future sample was used for all time steps. The results indicate better performance for the Solcast data, as this dataset contains no missing values.

With regard to the 95 % confidence interval, represented by the blue shaded region, the target values tend to fall outside this interval during the afternoon hours in both datasets. These findings suggest that applying the same window configuration across all time periods may not be appropriate for irradiance prediction. Therefore, a more detailed analysis on an hourly basis is recommended to improve prediction accuracy.

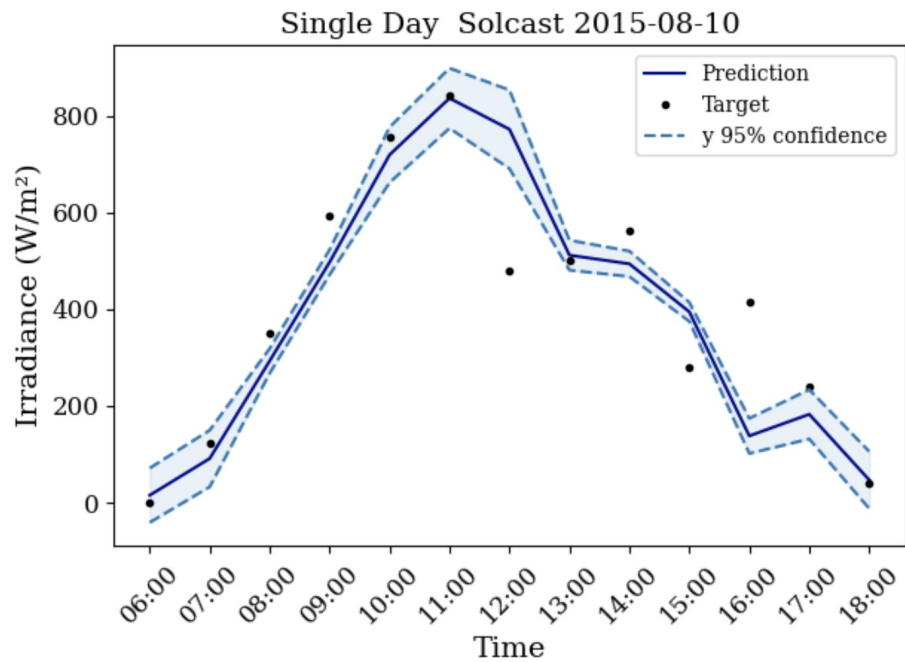


Figure 3. Predictions in the Solcast dataset using the LSTM model.

Source: Authors.

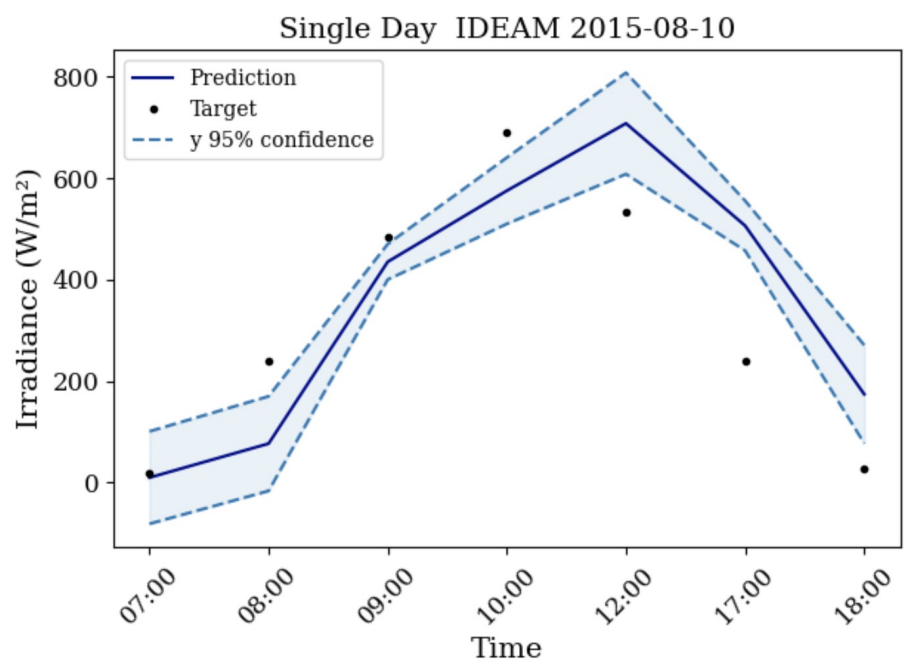


Figure 4. Predictions in the IDEAM dataset using the LSTM model

Source: Authors.

Conclusions

The research evaluated the performance of three different models — two based on recurrent neural networks and one on Gaussian processes — across three meteorological time series under three sliding-window configurations. This analysis made it possible to identify the most suitable model and configurations that best adapt to the intrinsic characteristics of the analyzed time series.

According to the results, in terms of accuracy, the SGPR model provided the best fit to the analyzed time series. In contrast, the GRU model yielded the lowest performance across the available datasets. Moreover, the hourly configuration improved prediction accuracy, as it considered only the most relevant information. Specifically, under the hourly scheme, SGPR achieved for the Jena dataset a MAE of 0.53 °C, an RMSE of 0.74 °C and an R^2 of 0.99 outperforming both LSTM (MAE = 0.74 °C, RMSE=0.98°C and R^2 = 0.99) and GRU (MAE = 0.89 °C, RMSE=1.18 °C, R^2 = 0.98).

Despite the method's strong performance across the evaluated datasets, some limitations remain to be addressed in future research. First, the presence of missing data in the IDEAM dataset disrupts temporal continuity; therefore, advanced models such as Generative Adversarial Networks (GANs) could be employed to improve data reconstruction. Second, the model's generalization to other regions of Nariño remains uncertain, and techniques such as transfer learning could be applied to extrapolate the model to new locations. Finally, the potential influence of additional meteorological variables on temperature or irradiance predictions has yet to be explored, which could be achieved through the incorporation of attention mechanisms.

References

- [1] K. Qadeer, A. Ahmad, M. A. Qyyum, A.-S. Nizami, and M. Lee, "Developing machine learning models for relative humidity prediction in air-based energy systems and environmental management applications," *Journal of Environmental Management*, vol. 292, p. 112736, 2021, <https://doi.org/10.1016/j.jenvman.2021.112736>.
- [2] E. Sarvas, N. Dimitropoulos, V. Marinakis, Z. Mylona, and H. Doukas, "Transfer learning strategies for solar power forecasting under data scarcity," *Scientific Reports*, vol. 12, pp. 1–13, 2022, <https://doi.org/10.1038/s41598-022-18516-x>.
- [3] İ. F. Şener and İ. Tuğal, "Optimized cnn-lstm with hybrid metaheuristic approaches for solar radiation forecasting," *Case Studies in Thermal Engineering*, p. 106356, 2025, <https://doi.org/10.1016/j.csite.2025.106356>.

- [4] C.-C. Lee, B. Zhou, T.-Y. Yang, C.-H. Yu, and J. Zhao, "The impact of urbanization on CO₂ emissions in China: The key role of foreign direct investment," *Emerging Markets Finance and Trade*, vol. 59, pp. 451–462, 2023, <https://doi.org/10.1080/1540496X.2022.2106843>.
- [5] J. Gaboitaolelwe, A. M. Zungeru, A. Yahya, C. K. Lebekwe, D. N. Vinod, and A. O. Saulau, "Machine learning based solar photovoltaic power forecasting: a review and comparison," *IEEE Access*, vol. 11, pp. 40 820–40 845, 2023, <https://doi.org/10.1109/ACCESS.2023.3270041>.
- [6] L. Abualigah, R. A. Zitar, K. H. Almotairi, A. M. Hussein, M. Abd Elaziz, M. R. Nikoo, and A. H. Gandomi, "Wind, solar, and photovoltaic renewable energy systems with and without energy storage optimization: A survey of advanced machine learning and deep learning techniques," *Energies*, vol. 15, p. 578, 2022, <https://doi.org/10.3390/en15020578>.
- [7] B. Lim and S. Zohren, "Time-series forecasting with deep learning: a survey," *Philosophical Transactions of the Royal Society A*, vol. 379, p. 20200209, 2021, <https://doi.org/10.1098/rsta.2020.0209>.
- [8] W. Li and K. E. Law, "Deep learning models for time series forecasting: A review," *IEEE Access*, vol. 12, pp. 92 306–92 327, 2024, <https://doi.org/10.1109/ACCESS.2024.3422528>.
- [9] N. B. Mohamad, A.-C. Lai, and B.-H. Lim, "A case study in the tropical region to evaluate univariate imputation methods for solar irradiance data with different weather types," *Sustainable Energy Technologies and Assessments*, vol. 50, p. 101764, 2022, <https://doi.org/10.1016/j.seta.2021.101764>.
- [10] T. Niu, J. Li, W. Wei, and H. Yue, "A hybrid deep learning framework integrating feature selection and transfer learning for multi-step global horizontal irradiation forecasting," *Applied Energy*, vol. 326, p. 119964, 2022, <https://doi.org/10.1016/j.apenergy.2022.119964>.
- [11] A. Mendyl, B. Mabasa, H. Bouzghiba, and T. Weidinger, "Calibration and validation of global horizontal irradiance clear sky models against mcClear clear sky model in Morocco," *Applied Sciences*, vol. 13, p. 320, 2022, <https://doi.org/10.3390/app13010320>.
- [12] A. Behrouz, M. Santacatterina, and R. Zabih, "Chimera: Effectively modeling multivariate time series with 2-dimensional state space models," *Advances in Neural Information Processing Systems*, vol. 37, pp. 119 886–119 918, 2024, https://proceedings.neurips.cc/paper_files/paper/2024/file/d8e80772c27beff4ae1676fb147bbf26-Paper-Conference.pdf.
- [13] A. Dikshit, B. Pradhan, and A. M. Alamri, "Long lead time drought forecasting using lagged climate variables and a stacked long short-term memory model," *Science of The Total Environment*, vol. 755, p. 142638, 2021, <https://doi.org/10.1016/j.scitotenv.2020.142638>.

- [14] H. Yadav and A. Thakkar, "Noa-lstm: An efficient lstm cell architecture for time series forecasting," *Expert Systems with Applications*, vol. 238, p. 122333, 2024, <https://doi.org/10.1016/j.eswa.2023.122333>.
- [15] L. S. Hoyos-Gómez, J. F. Ruiz-Muñoz, and B. J. Ruiz-Mendoza, "Short-term forecasting of global solar irradiance in tropical environments with incomplete data," *Applied Energy*, vol. 307, p. 118192, 2022, <https://doi.org/10.1016/j.apenergy.2021.118192>.
- [16] A. Agga, A. Abbou, M. Labbadi, Y. El Houm, and I. H. O. Ali, "Cnn-lstm: An efficient hybrid deep learning architecture for predicting short-term photovoltaic power production," *Electric Power Systems Research*, vol. 208, p. 107908, 2022, <https://doi.org/10.1016/j.epsr.2022.107908>.
- [17] H. Zhang, B. Li, S.-F. Su, W. Yang, and L. Xie, "A novel hybrid transformer-based framework for solar irradiance forecasting under incomplete data scenarios," *IEEE Transactions on Industrial Informatics*, vol. 20, pp. 8605 – 8615, 2024, <https://doi.org/10.1109/TII.2024.3369671>.
- [18] A. Almarshoud *et al.*, "Validation of satellite-derived solar irradiance datasets: a case study in saudi arabia," *Future Sustainability*, vol. 2, no. 2, pp. 1–7, 2024, <https://doi.org/10.55670/fpll.fusus.2.2.1>.
- [19] A. M. Assaf, H. Haron, H. N. Abdull Hamed, F. A. Ghaleb, S. N. Qasem, and A. M. Albarrak, "A review on neural network based models for short term solar irradiance forecasting," *Applied Sciences*, vol. 13, no. 14, p. 8332, 2023, <https://doi.org/10.3390/app13148332>.
- [20] W. Dang, S. Liao, B. Yang, Z. Yin, M. Liu, L. Yin, and W. Zheng, "An encoder-decoder fusion battery life prediction method based on gaussian process regression and improvement," *Journal of Energy Storage*, vol. 59, p. 106469, 2023, <https://doi.org/10.1016/j.est.2022.106469>.
- [21] G. Etxegarai, A. López, N. Aginako, and F. Rodríguez, "An analysis of different deep learning neural networks for intra-hour solar irradiation forecasting to compute solar photovoltaic generators' energy production," *Energy for Sustainable Development*, vol. 68, pp. 1–17, 2022, <https://doi.org/10.1016/j.esd.2022.02.002>.
- [22] H. Tolba, N. Dkhili, J. Nou, J. Eynard, S. Thil, and S. Grieu, "Multi-horizon forecasting of global horizontal irradiance using online gaussian process regression: A kernel study," *Energies*, vol. 13, p. 4184, 2020, <https://doi.org/10.3390/en13164184>.

