forecasting

MDPI

Article

Methodology Based on BERT (Bidirectional Encoder Representations from Transformers) to Improve Solar Irradiance Prediction of Deep Learning Models Trained with Time Series of Spatiotemporal Meteorological Information

Llinet Benavides-Cesar, Miguel-Ángel Manso-Callejo * and Calimanut-Ionut Cira

Departamento de Ingeniería Topográfica y Cartográfica, Escuela Técnica Superior de Ingenieros en Topografía Geodesia y Cartografía, Universidad Politécnica de Madrid, C/Mercator 2, 28031 Madrid, Spain; llinet.bcesar@upm.es (L.B.-C.); ionut.cira@upm.es (C.-I.C.)

* Correspondence: m.manso@upm.es

Abstract: Accurate solar resource forecasting is important because of the inherent variability associated with solar energy and its significant impact on the cost for energy producers. The traditional method applied in solar irradiance forecasting involves two main phases, related to (1) data selection and (2) model selection, training, and evaluation. In this study, we propose a novel end-to-end methodology for solar irradiance forecasting that starts with the search for the data and all of the preprocessing operations involved in obtaining a quality dataset, continuing by imputing missing data with the BERT (Bidirectional Encoder Representations from Transformers) model, and ending with obtaining and evaluating the predicted values. This novel methodology is based on three phases; namely, Phase_1, related to the acquisition and preparation of the data, Phase_2, related to the proposed imputation with a BERT model, and Phase_3, related to the training and prediction with new models based on deep learning. These phases of the proposed methodology can be applied in a disjointed manner, and were used on two public datasets accessible to the scientific community. Each of the proposed phases proved to be valuable for the workflow, and the application of the novel method delivered increases in performance of up to 3 percentage points (3%) when compared to the traditional approach.

Keywords: solar forecast method; BERT; deep learning; data imputation

check for updates

Academic Editor: Ted Soubdhan

Received: 18 November 2024
Revised: 1 January 2025
Accepted: 14 January 2025
Published: 17 January 2025

Citation: Benavides-Cesar, L.; Manso-Callejo, M.-Á.; Cira, C.-I. Methodology Based on BERT (Bidirectional Encoder Representations from Transformers) to Improve Solar Irradiance Prediction of Deep Learning Models Trained with Time Series of Spatiotemporal Meteorological Information. *Forecasting* **2025**, *7*, 5. https://doi.org/10.3390/forecast7010005

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

For solar forecasting, the standard methodology applied for training deep learning models is generally based on two fundamental phases, (1) data selection (phase where the solar irradiance information required for training the models is obtained) and (2) model selection, training, and evaluation (phase where the machine learning models relevant for the task are selected, trained, and optimized, and where the method for assessing the performance of the trained models is selected and performed) [1].

The selection of the dataset begins with its collection, and this initial step is heavily influenced by the available data sources, particularly whether they are public or private [2], or whether the data will be obtained from a web service, a repository, or will be generated. It is also important to assess the temporal resolution required (based on the requirements of the energy production prediction [3]). Finally, the period (e.g., the number of years of the dataset) and its spatial resolution are also important aspects. This initial step also includes operations of data analysis and evaluation, where it is determined if there are missing data,

if there are values outside the established limits, or if the temporal index is complete; some of these problems can be solved by cleaning the data. In the case of missing data, data imputation can be performed depending on whether the data loss is significant.

It is also important to note that the two phases of the classical methodology are highly correlated, as variation in one will influence how the other two are processed. For example, if a single time series is used, a classical statistical method such as an ARIMA can be considered as sufficient [4]. However, when multiple time series, satellite images, or sky images are processed, the model must feature a higher complexity to be capable of extracting the information from this type of data. In these cases, the use of machine learning or deep learning models is recommended. Finally, for the evaluation step, it would be necessary to choose between a deterministic or probabilistic approach, choosing the most used metrics according to the approach. This methodology becomes more complex when we analyze each of the steps and the processes to be developed within them (Figure 1).

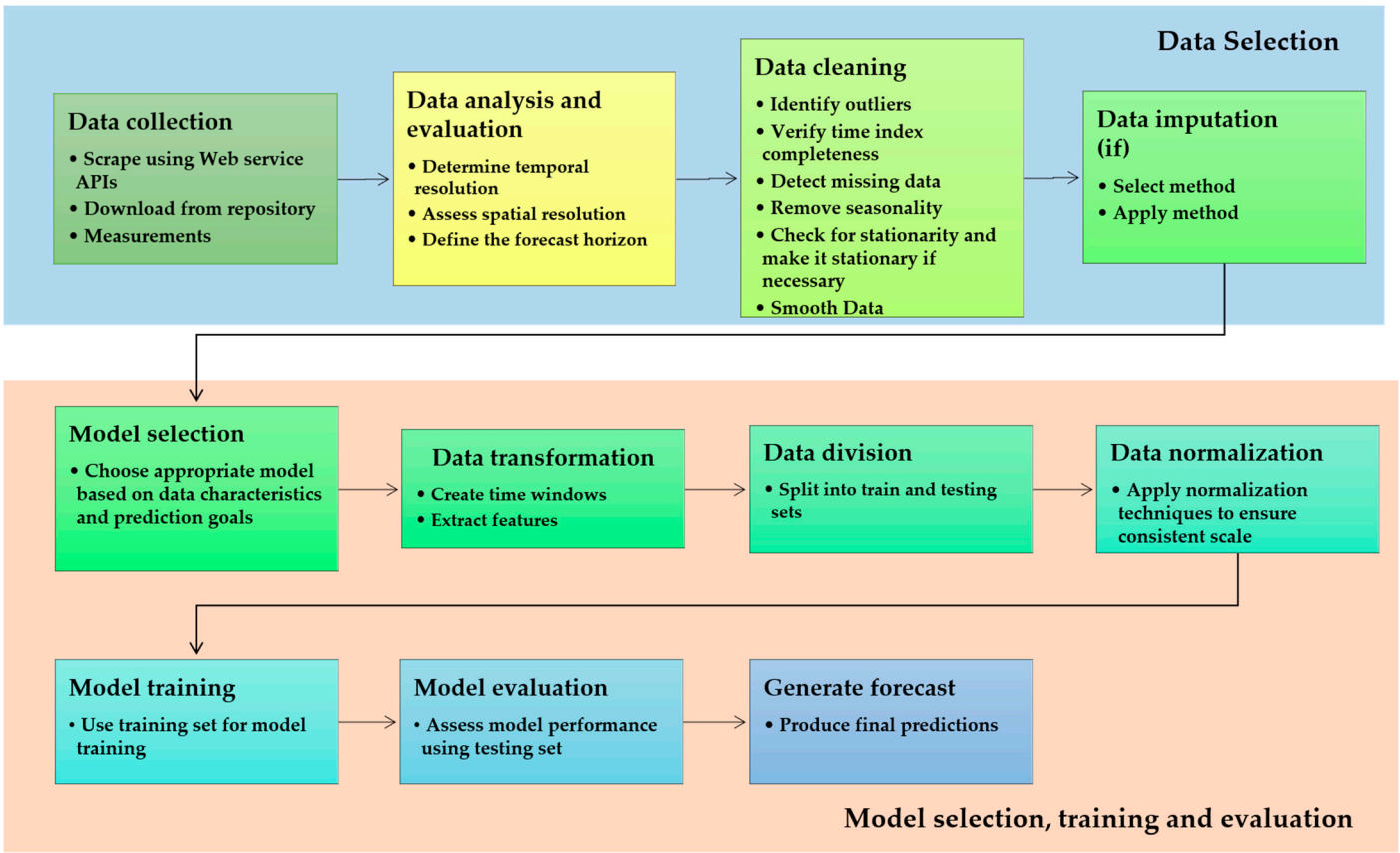


Figure 1. Diagram of the extended, standard methodology applied for solar irradiance prediction using meteorological time series.

Regarding the model selection step, a decision between physical or statistical ones is generally performed. Within the statistical ones, classical methods (ARIMA, Ridge, LASSO) or automated learning methods can be selected. Within the automated learning methods, there are also classical automated learning methods and deep learning methods. The model selection will also depend on the data, number of variables, size of the dataset, and data format. After selecting the model, the data must be transformed according to the input expected by the model—normally, time series forecasting is performed in a window; that is, given past data in time-steps, a model will predict one or more time-steps in the future. The size of the time window governs the amount of data considered when making the prediction. After obtaining the time windows, the data are divided into training

and test sets (validation sets can also be obtained where a large volume of information is available). This division should be performed considering that the evaluation data can never be part of the training. Additionally, due to the seasonality of the data, if possible, it is recommended to use whole years for the evaluation; if not possible, ensure a representation of the different climatic seasons. For data normalization, many authors [5–9] use the classic automated learning methods such as standard scaling to transform the data to have a mean (μ) of 0 and a standard deviation (σ) of 1 (generally the values will fall within the $[-3, 3]$ range, considering a normal distribution), or the min–max scaling to translate the data to a specified range (usually, [0, 1], particularly useful for model implementations that are sensitive to the scale of data, such as neural networks), although the recommended method for solar radiation prediction is to use clearness or clear sky indices [10]. In addition to normalizing, it is also recommended to eliminate irrelevant information for the model to improve the quality of the prediction.

The methodology proposed in this work introduces a new imputation model in the processing of missing data, so the next section is dedicated to further explain this topic.

Imputation in Solar Forecasting

In solar energy forecasting, as in many other fields of study, it is important to have complete and accurate data for analysis and decision-making processes, and decisions regarding the imputation of missing data are a common challenge. This challenge becomes even more relevant in time series analysis, where data continuity and order are critical, time series being sequences of data that are recorded at regular time intervals and are ordered according to the time at which they are recorded [11].

The imputation method is an important analysis in time series and involves "filling in" missing values in the series using the available observed values [12]. In this regard, imputation of solar power time series [13] is a topic that has been explored with traditional statistical methods [14,15] and with more modern automated learning methods [16,17]. Demirhan et al. [15] evaluated 36 imputation methods for solar irradiance series with a dataset collected in Australia; the imputation methods considered were variants of the methods listed below, namely, (1) interpolation (such as linear, spline, or Stineman), (2) Kalman filters, (3) persistence, (4) weighted moving average, and (5) random sampling. The authors designed sixteen experimental scenarios and determined that linear interpolation and Stineman interpolation—methods that utilize a function passing through a series of points in the xy plane to estimate slopes—proved to be the most accurate for minute and hourly series. For daily and weekly series, however, the weighted moving average method yielded the best results.

De-Paz-Centeno et al. [17] proposed a neural network to impute values for series with missing values in ranges from 30% to 70% of the total number of values, and recommended its use for scenarios with 50% missing values. The proposed model was a convolutional neural network employing an encoder–decoder architecture, tested on both a private and a public dataset, each comprising two years of samples. This architecture achieved coefficients of determination between 0.81 and 0.98, significantly outperforming the other models evaluated.

Given the sequential characteristics of time series data, models originally designed for Natural Language Processing (NLP), such as transformers, may also be well-suited for this task. Transformers are based on attentional mechanisms (relating different positions of the same sequence to compute a representation of it; also known as self-attention), have proven effective in solving sequential tasks, and have been applied to and have obtained good results in time series imputation [18,19], while easily handling long-range dependencies [20].

Bidirectional Encoder Representations from Transformers (BERT) [21] is a transformer-based deep learning model that uses bidirectional self-attention by jointly conditioning the left and right context, being one of the most popular deep learning-based language models [22]. BERT is pre-trained using two key unsupervised tasks, the Mask Language Model (MLM) and Next Sentence Prediction (NSP). In MLM, some words in the input sequence are randomly masked, and the goal is to predict the masked words based solely on their context. The NSP task, on other hand, focuses on text-pair representations to predict a sentence from the previous sentence. BERT has also been pre-trained for other knowledge areas, such as computer vision [23,24], bioinformatics, and computational biology [25–27], or learning geospatial representations based on a point of interest [28].

In the methodology proposed in this work, the BERT model was applied for irradiance time series imputation [29] by training it from scratch for the MLM task with irradiance data. The methodology also makes use of three novel deep learning methods based on convolutional networks for solar forecasting [30].

The main contributions of this research are summarized as follows:

- An end-to-end methodology based on the BERT model that contains a complete process (from data gathering to the final forecast) and is applicable for solar forecasting workflows to improve the quality of the prediction results.
- To prove its applicability, the proposed methodology is implemented by training three deep learning models on the CyL-GHI dataset (containing data from Castile and Léon, Spain), and on a dataset established in the literature (with data from Hawaii, USA). The application of the method on the mentioned datasets delivered improvements in performance of up to 3%.

The rest of this paper is organized as follows. Section 2 presents an end-to-end methodology. To evaluate the robustness of the proposed method, Section 3 details the application of the methodology on the CyL-GHI dataset (obtained as a result of applying a part of the proposed methodology), while Section 4 presents the application of the methodology on a public dataset from Hawaii, USA. This section includes the assessment of the results with statistical analyses. Section 5 discusses the strengths and limitations of the proposed methodology Finally, Section 6 presents the conclusions reached.

2. Proposed End-to-End Methodology

In this section, the proposed methodology to improve the solar irradiance prediction with a time series of meteorological observations enriched with spatiotemporal data is described. The proposed methodology is presented in Figure 2 and starts with processing the raw data to obtain the training data. Afterwards, it includes steps related to the model selection, imputation, and training. The end-to-end method then continues with applying feature selection methods [30], and also includes two steps related to imputation [29] and stationarization [10].

2.1. Phase_1: Obtaining the Dataset

The first phase of the proposed methodology is related to the steps to be applied for obtaining a dataset of high quality, and is based on the method introduced in Ref. [2].

In this process, it is important to first define the source of the data (depending on the prediction horizon and temporal resolution chosen) and define and how it can be accessed. After obtaining the data, one can begin analyzing the data for inconsistencies. The series should be analyzed to ensure that the sequence of dates is correct without gaps, and that the format of presentation of the dates is as expected (verify time index completeness).

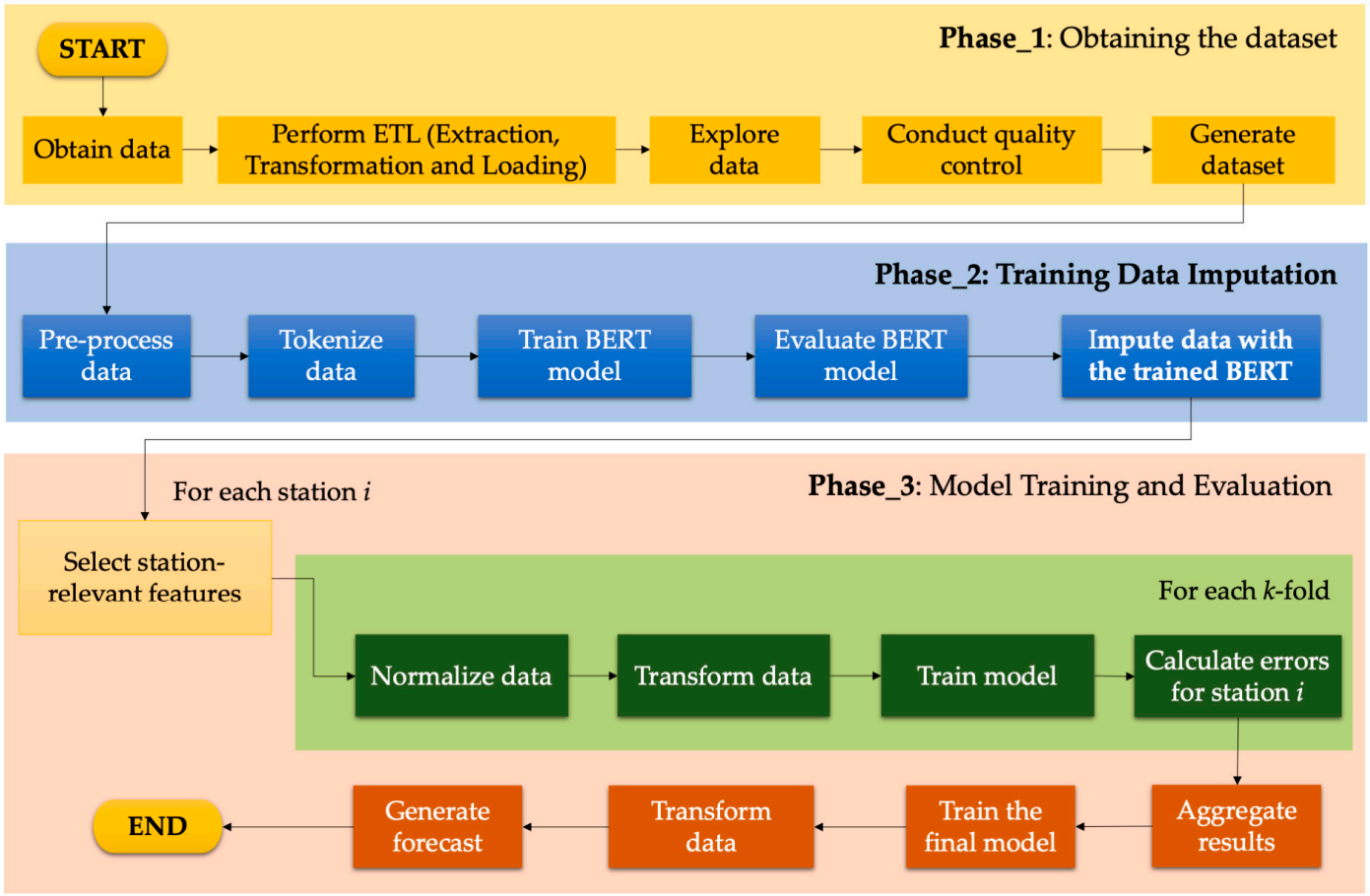


Figure 2. A proposed end-to-end methodology for solar irradiance prediction based on BERT for data imputation and deep learning models trained with a time series of spatiotemporal meteorological information. Notes: (1) The proposed method is represented as a flowchart to be applied from the initial steps of data collection to the final stages of forecast generation. (2) Sections 2.1–2.3 provide details regarding the implementation of the three proposed phases of the methodology.

Due to limitations in access to information from quality weather stations, it is important to complement information with data from repositories or web services that provide information collected from satellites or generated by mathematical models. These data sources can complement irradiance data with atmospheric, astronomical, or meteorological variables that help to better understand the environment in which the prediction is being made. The challenge with using different data sources is that often the delivery format is not homogeneous. To enrich the information using data in different formats, Extraction, Transformation, and Loading (ETL) techniques can be applied to unify all of the information in a single file or database featuring the same data format.

When working with several meteorological stations, one should evaluate whether the same features are obtained from all of them. In the case of not obtaining the same ones, one can discriminate according to the specific objectives, and determine which features have more weight. It is also important to review for outliers and define how to handle them, check the format of the data, or apply decomposition into a simpler format when needed, leading to new features for the dataset (for example, from the date, derive the day of the year or the year). It is also important to detect missing data and to smooth the data.

The exploratory data analysis should be carried out to identify problems or inconsistencies in the dataset. It can be done by graphically representing the information, or by using libraries that automatically compute relevant data statistics.

After data analysis and the related data cleaning that is usually involved, one can perform quality control following the protocols according to the variables to be evaluated. In the case of quality control, it is also important to be aware that there are specific controls depending on what is tested—each variable can even have its own quality control. Therefore, it is always advisable to review the literature looking for controls already established and standardized by other authors. In a general perspective, it is recommended to always attempt to ensure the accuracy, completeness, and consistency of the data.

When the dataset passes quality control, a dataset considered the end result of the cleaning and quality assessment process is generated. If this generated dataset is complex or presents loss of data that requires the need for an imputation process, Phase_2 of the methodology will be applied; Phase_2 of the proposed methodology is explained in the next section.

2.2. Phase_2: Training Data Imputation

A more detailed explanation of the missing data imputation with BERT [29] (Phase_2 of Figure 2, represented with a blue rectangle) is presented as a pseudocode form in Algorithm 1. As for the notations used in Algorithm 1, E refers to the irradiance values collected by a meteorological station, D_{train} and D_{test} represent the division into training and test set of each dataset, respetively, d_f denotes the days with missing values, while "trained model BERT" reference the new imputation model obtained after using irradiance values to train a BERT model from scratch.

Algorithm 1: Data Processing and Imputation Procedure (Phase_2)

Input: Dataset D, last test year, imputation model
Output: Dataset with imputed values

For each station E in the set of stations **do**:
    Filter D_E to retain only values where solar elevation > 5
    Split D_E into training set D_{train} (years < last year) and test set D_{test} (last year)
    Remove days from D_{train} and D_{test} with missing values
    For each day d € D_{train} and D_{test} **do**
        Convert d into a sequence of values
        Save the sequence of values of d into a text file
    End for
End for
Tokenize the text files generated in the previous step
Train the imputation model using tokenized D_{train}
Evaluate the model using D_{test}
Save new imputation model
For each day with missing values d_f € D **do**
    Convert d_f into text format
    Impute missing values in d_f using the trained model BERT
    Convert the imputed series from text back to time series format
End for
Save the dataset with imputed values

The pseudocode of Algorithm 1 specifies that, for each station shown, only values of irradiance with solar elevation greater than 5° [31] are considered (eliminating nighttime values and values close to sunrise or sunset).

For preprocessing, the dataset of each station is separated into training and test sets (leaving the last year for testing), and only days with no missing information should be

used. For each day of the dataset, the one-day time series is converted into a sequence of values. A text file where each row represents the values of one day is created.

Afterwards, the Tokenization operation is performed for the files obtained in the previous step to convert text data into numerical tokens that can be processes by DL models. For training BERT, a vocabulary of 1600 tokens (conformed to the values that the irradiance can reach) was used. A series of special tokens are used, (1) [CLS] to identify the beginning of the sequence, (2) [MASK] to mask the position of the chain we want to predict, (3) [UNK] to point out the elements of the sequence that are unknown, (4) [PAD] to fill sequences that are shorter than those expected by the model, and (5) [SEP] used to separate two sequences. The configuration of parameters used to obtain the best model features a batch size of 64, 12 hidden layers, and 12 attentions heads.

For the evaluation of the model, the test period (for example, one year) that was separated earlier should be used to compute relevant performance metrics such as Mean Absolute Error (MAE), the Root Mean Squared Error (RMSE), and the forecast skill score (FS) defined by Equations (1)–(3), respectively.

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |(y_i - \hat{y}_i)|}{n} \tag{1}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \tag{2}$$

$$\mathrm{FS} = 1 - \frac{\mathrm{RMSE}_{\mathrm{model}}}{\mathrm{RMSE}_{\mathrm{persistence}}} \tag{3}$$

The model should perform well in predicting consecutive missing values in a sequence and be able to replicate the irradiance distribution curve of the day before being used for missing data imputation.

Once the new imputation model is obtained, the imputation of the series with missing values is performed by first converting the available series to text format. To obtain the imputed data, the desired position is masked, and the model is queried (this is done recursively until all of the unknown values in the string have been obtained). Afterwards, the text is transformed into the time series format again. As a final step, a new imputed training dataset is generated, improving the dataset's completeness.

2.3. Phase_3: Model Training and Evaluation

The final phase of the end-to-end methodology is based on the feature selection methods proposed in Ref. [30], and also includes the steps related to stationarization [10].

Phase_3 begins with defining the subset of variables to be used as input to the model. In this process, feature selection methods varying from applying correlation analysis to determining the feature importance automatically (for example, identifying the features with the highest impact on the predictions with the feature importance scores of tree-based models) together with the criteria of experts in the field are employed.

As input to the models, the GHI of the target station and neighboring stations, as well as the solar elevation and azimuth angles, should be selected, as using information from neighbors benefits short-term forecasting with data on cloud movements. In this step, it is important to define the spatial and temporal correlation zone with its close neighbors. A sliding window structure is used to divide the time series into overlapping segments of fixed size, giving as output the matrix to be used by the models.

It is also important to remove seasonality and check for stationarity and stationarize the time series (if necessary). The irradiance should be normalized using the clear sky index and the data transformed into the structure expected by the models. After defining the

characteristics, state-of-the-art models should be trained *k*-fold cross-validation to adjust the parameters until the ideal configuration is reached.

To assess the performance of the model, *k*-fold cross-validation can be used. In this case, the model is trained *k* times, each time using a different combination of $k-1$ subsets for training and the remaining subset for testing. The *k*-fold cross-validation provides a robust and unbiased assessment of model performance. The performance metric from all *k* iterations should be averaged to provide a single overall evaluation score. This aggregation ensures that the model is tested on different subsets of data, reducing the risk of overfitting and providing a more complete understanding of its generalization capabilities.

The final model is trained with the complete training data and evaluated with the separate test set. For this training, the insight gained from the cross-validation is used to ensure that the model is fitted to obtain its best performance. By using it on the complete set, its forecasting skill is maximized. For the final forecast, the obtained value must be multiplied again by the respective clear sky irradiance, as the goal is to predict the irradiance value again.

The obtained model can be stored in a format such as H5 for interoperability. To generate predictions from the trained model, the 'predict' method, which requires the same preprocessing steps and the same environment as during training, can be applied to ensure consistency. The result is a model used in production that receives real-time or batch data, processes it through the trained architecture, and generates predictions, providing practical information for decision making. An integration of the model with dashboards or APIs would further enrich the predictions and the experience of end users.

2.4. Experimental Design

The proposed end-to-end method is applied to two datasets in Sections 3 and 4, one proposed for the authors and obtained as a result of the application of this methodology (Section 3.1), and one established in the literature (described in Section 4.1); the unification of all of these procedures and operations improved the quality of the final predictions.

For the implementation of the methodology, Python programming language [32] was used to define the models with Keras [33] and Tensorflow [34] libraries. The libraries scikit-learn [35], pandas, NumPy [36], seaborn [37], and matplotlib [38] were also used for data manipulation and graph generation operations. The experiments were carried out on a server featuring two NVIDIA RTX3060 GPU cards with 8 and 12 GB of VRAM and an Intel i7 processor with a 1 TByte SSD disk.

The experiments were performed following the workflow proposed in Figure 2 to emphasize the two different perspectives of the process, data imputation and model selection. The three performances metrics used to assess the accuracy of the regression models in the following sections are the MAE, RMSE, and FS metrics (defined in Equations (1)–(3)).

The deep learning models evaluated are ST_CNN_v1, ST_CNN_v2, and ST_Dilated_CNN, proposed in Ref. [30]. The average run time per epoch for the ST_CNN_V1 model for a station is 121 s, for the ST_CNNN_v2 model it is 154 s, and for the ST_Dilated_CNN model it is 183 s. As the complexity of the model increases, the execution time becomes longer. Given the model architecture and the size of the dataset, the execution time can be considered competitive and reflects an efficient implementation.

The next two sections detail the application of the proposed methodology in both cases.

3. Implementation of the Proposed Methodology on the CyL-GHI Dataset

The methodology proposed in Section 2 can be applied in a disjointed manner, depending on the project. When starting from a published dataset that has already passed a quality check, Phase_1 is restricted to getting familiar with the dataset. If, in addition, the dataset

has no missing data, it is passed directly to Phase_3 without the need to use Phase_2. For example, Phase_1 (represented with a yellow rectangle) and Phase_2 (represented with a blue rectangle) of Figure 2 show the procedure applied to obtain and impute the missing data from the CyL-GHI dataset. Phase_3 (represented with the rectangle of desert sand color in Figure 2) was applied to all stations from CyL-GHI.

The deep learning ST_CNN_v1, ST_CNN_v2, and ST_Dilated_CNN models trained in this study are convolutional network-based models that were introduced in Ref. [30] for the spatiotemporal prediction of solar irradiance using exogenous variables.

3.1. Data: CyL-GHI Dataset

The data used for the first implementation of the proposed methodology belong to the CyL-GHI dataset that was presented in Ref. [2]. In particular, two stations (P03 and SO01, Fuentes de Nava and Almazán, respectively, from two provinces of the Castile and León region, Spain) were selected from the subset of stations used in Ref. [30]. Figure 3 shows the map of stations in the region, highlighting the stations considered in this study.

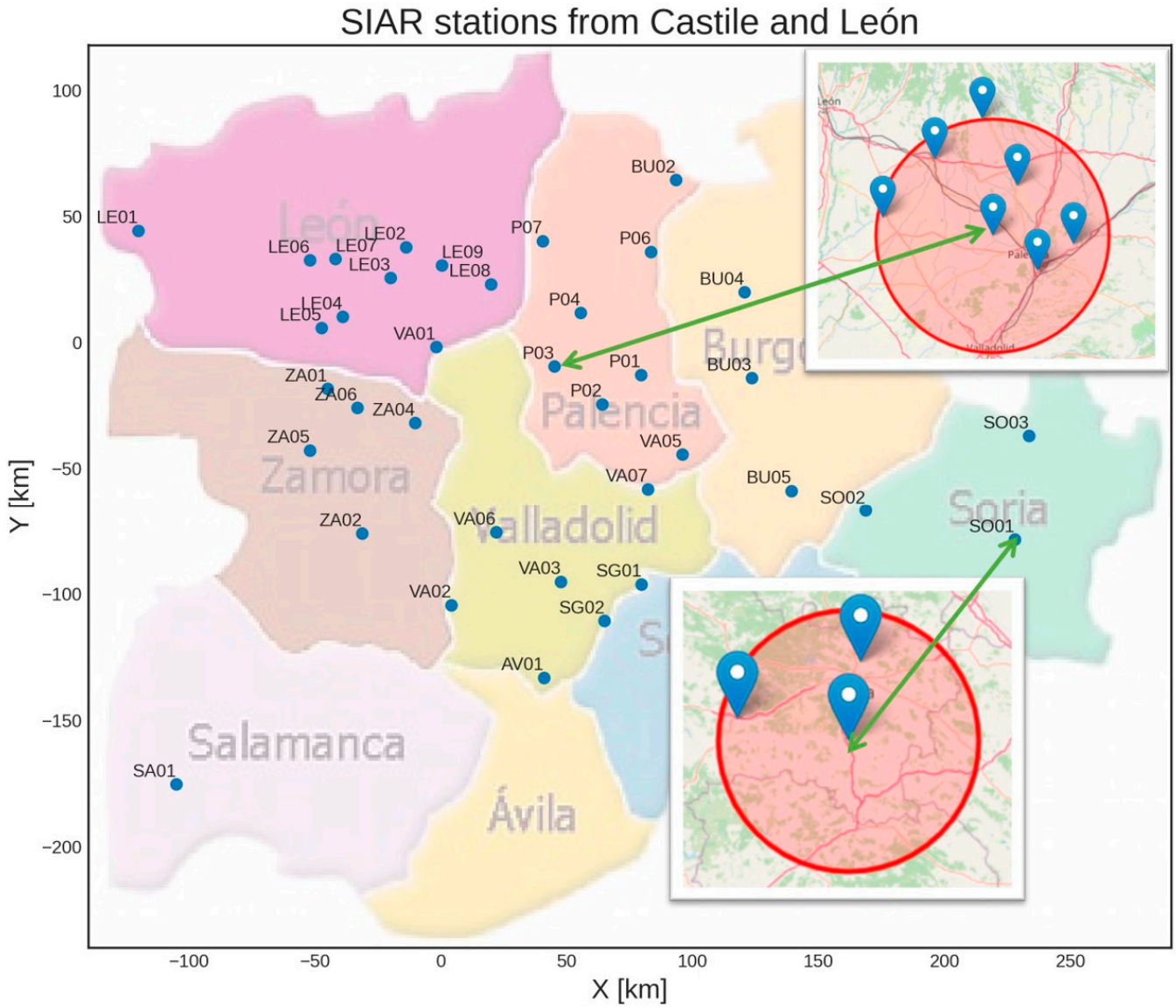


Figure 3. Spatial representation of the stations selected for the experiments, with P03 belonging to Fuentes de Nava (province of Palencia) and SO01 to belonging to Almazán (province of Soria), within the Castile and León region (Spain). LEn° identifies stations in the province of León, as well as SAn° Salamanca, ZAn° Zamora, Pn° Palencia, SGn° Segovia, SOn° Soria and BUn° Burgos.

3.2. Results

The results achieved from applying the methodology proposed in Figure 2 by the models proposed in Ref. [30] are compared for two stations in Table 1. Table 1 shows the results achieved with imputation. Comparing the MAE for all models proves that working with imputed data decreases the error for all models. The distinction is that for station SO01, the changes are more noticeable than for station P03. In contrast, while the RMSE of station SO01 improves for all models, station P03 shows no significant improvement.

Table 1. Comparative results between the evaluation of the trained models (ST_CNN_v1, ST_CNN_v2, and ST_Dilated_CNN) with imputed and non-imputed data.

Model	Training Data	Performance Metric	Station	
			P03	**SO01**
ST_CNN_v1	Not imputed	FS (%)	26.05	18.03
		MAE (W/m^2)	40.4	41.18
		RMSE (W/m^2)	69.31	68.81
	Imputed	FS (%)	26.53	29.65
		MAE (W/m^2)	39.19	32.6
		RMSE (W/m^2)	68.86	59.05
ST_CNN_v2	Not imputed	FS (%)	27.47	27.27
		MAE (W/m^2)	41.06	38.15
		RMSE (W/m^2)	67.97	61.05
	Imputed	FS (%)	27.45	30.65
		MAE (W/m^2)	38.26	31.1
		RMSE (W/m^2)	67.99	58.21
ST_Dilated_CNN	Not imputed	FS (%)	27.39	28.32
		MAE (W/m^2)	40.81	40.46
		RMSE (W/m^2)	68.05	60.17
	Imputed	FS (%)	27.26	32.42
		MAE (W/m^2)	37.83	30.49
		RMSE (W/m^2)	68.17	56.72

Since the FS is linked to the RMSE error, its behavior in the models is similar to that found with the RMSE. The RMSE for station SO01 has values with differences of 8 pp. Station SO01 benefits the most from imputation because it improves its results for all models. The MAE metric decreases for all models when working with imputed data. This shows that the imputation phase based on the introduction of a BERT model delivered increases in performance.

Figure 4 shows a comparison between the data obtained for stations P03 and SO01 with imputed and non-imputed data. Apart from the ST_Dilated_CNN model for station P03, all of the FS values are higher when trained on imputed data. The differences are more marked at station SO01 for all models. The ST_CNN_v2 and ST_Dilated_CNN models are the models with the best FS values, showing their forecast skill.

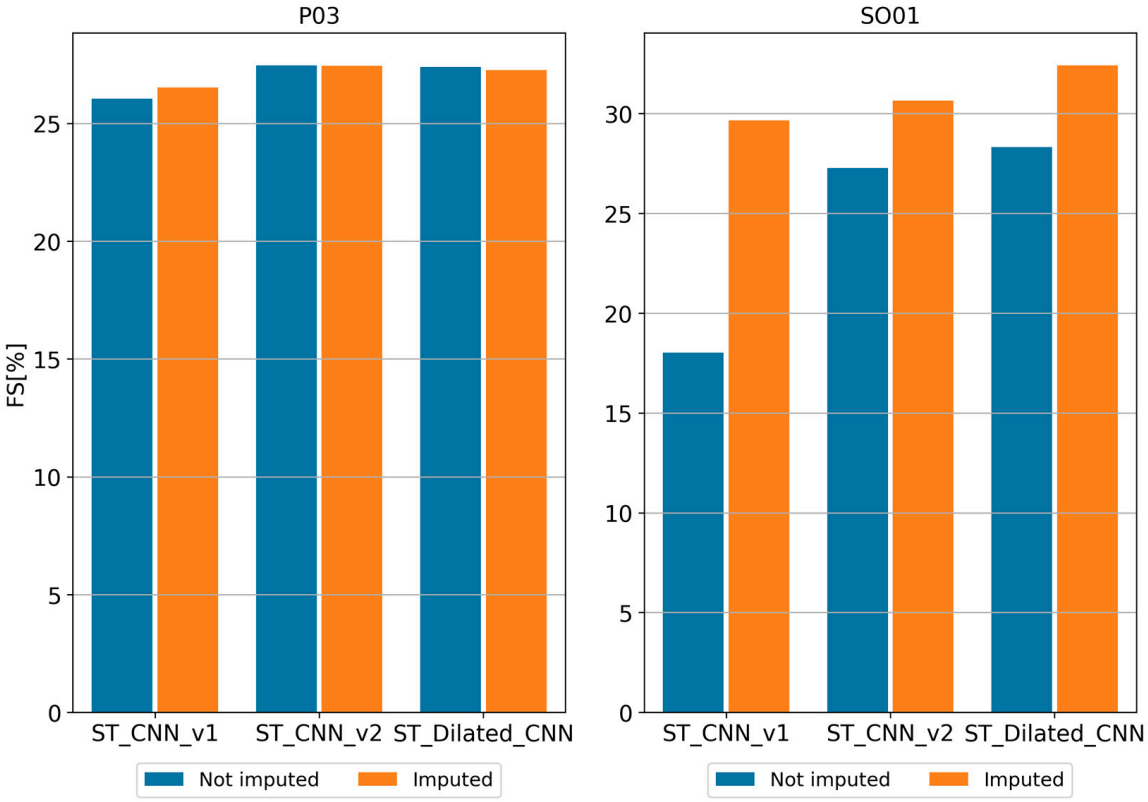


Figure 4. Comparison of FS metrics with and without imputed data for models ST_CNN_v1, ST_CNN_v2, and ST_Dilated_CNN for stations P03 and SO01.

These particular stations had the same percentage of missing data [2]. However, it can be observed that station P03 does not benefit equally from the imputation (Figure 4) when a more detailed analysis is performed considering the input data. In the case of station SO01, it receives, among others, data from station SO03, which has a higher loss of data, and which benefits more from the imputation performed in Phase_2. As more data are available, the results obtained are better at this station. Stations close to P03 have a lower percentage of information loss, so their imputation leads to lower gains in the final input to the models.

3.3. Analysis of the BERT Imputation on the CyL-GHI Dataset

In the imputation process of the CyL-GHI dataset, the BERT model was trained from scratch. The MLM task was employed to impute missing data. Two scenarios were evaluated: the first, where a single missing value at a specific position in the sequence is imputed, and a second scenario, where a specific position in the sequence is imputed taking into account that all other values in the sequence from that position onwards are unknown.

The second scenario is the most interesting because when performing imputation with a linear interpolation, if from one point the remaining values are not known (i.e., the data for the remaining points are missing), the interpolation cannot continue beyond the last known value (Appendices A and B). This new model offers a different solution for these cases. In this section, when discussing unknown values, this particular case is referred to.

In the case of figures that have the reserved 'mask' label, this refers to the position to be filled, according to the training of the model.

Figure 5 shows the distribution of the RMSE for each of the stations when using known values. For most stations, the median RMSE remains between 70 and 90, suggesting that the model performs well overall. However, some stations, such as station BU02, show a higher variability in the errors, which is reflected in their representation, indicating a less consistent performance at that station.

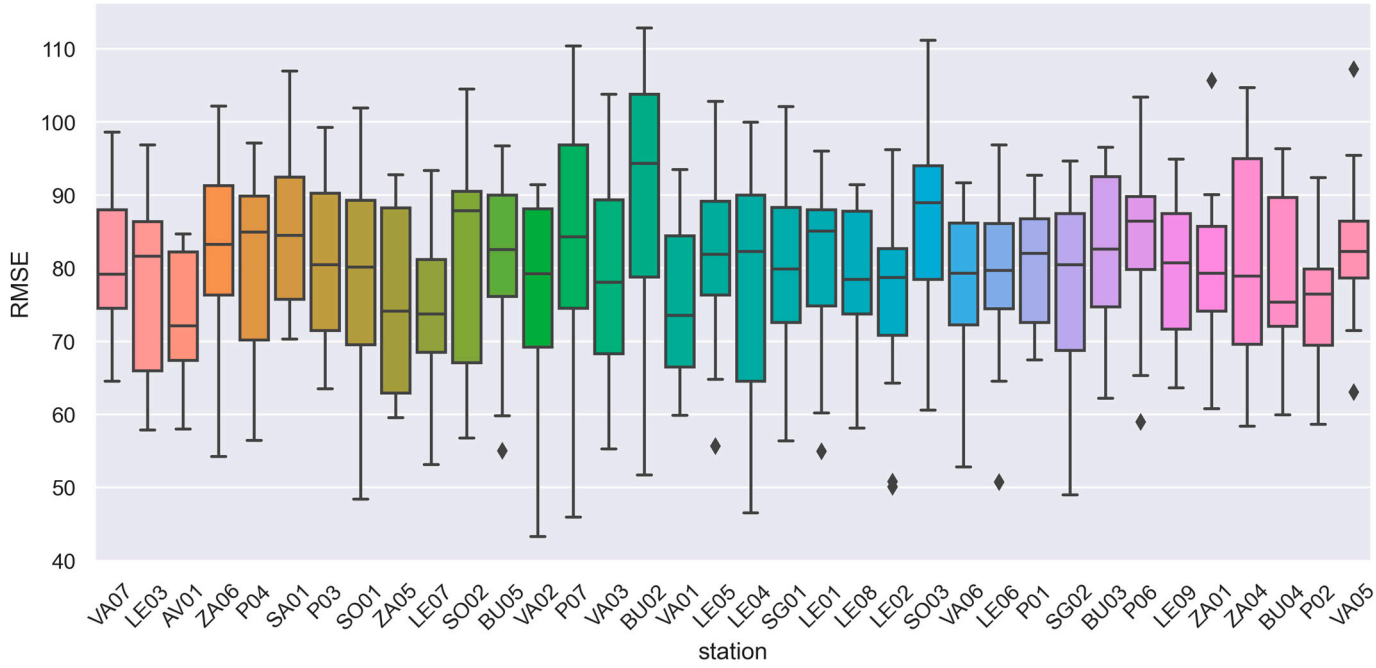


Figure 5. RMSE distribution for each of the stations, where symbol ♦ represents extreme values.

In addition, some outliers are observed, particularly at stations ZA01 and VA05, suggesting that the model had higher prediction errors at certain times. These cases may indicate the need for further investigation of the particular conditions at these stations. In summary, the box plot suggests that, although the model predicts well at several stations, there are some stations with higher variability or outlier errors, which may require more specific adjustments or approaches.

Figure 6 shows the distribution of the MAE. For most stations, the average MAE is in a range close to 45, suggesting that the model performs similarly in terms of absolute error. However, the behavior observed in the RMSE plot is repeated for stations with outliers, suggesting that in those cases the model had larger errors for some specific points. This may be due to data conditions at those stations. In summary, the graph indicates that, in general, the model has a stable performance, but there are stations where the errors tend to vary more or present punctually high errors.

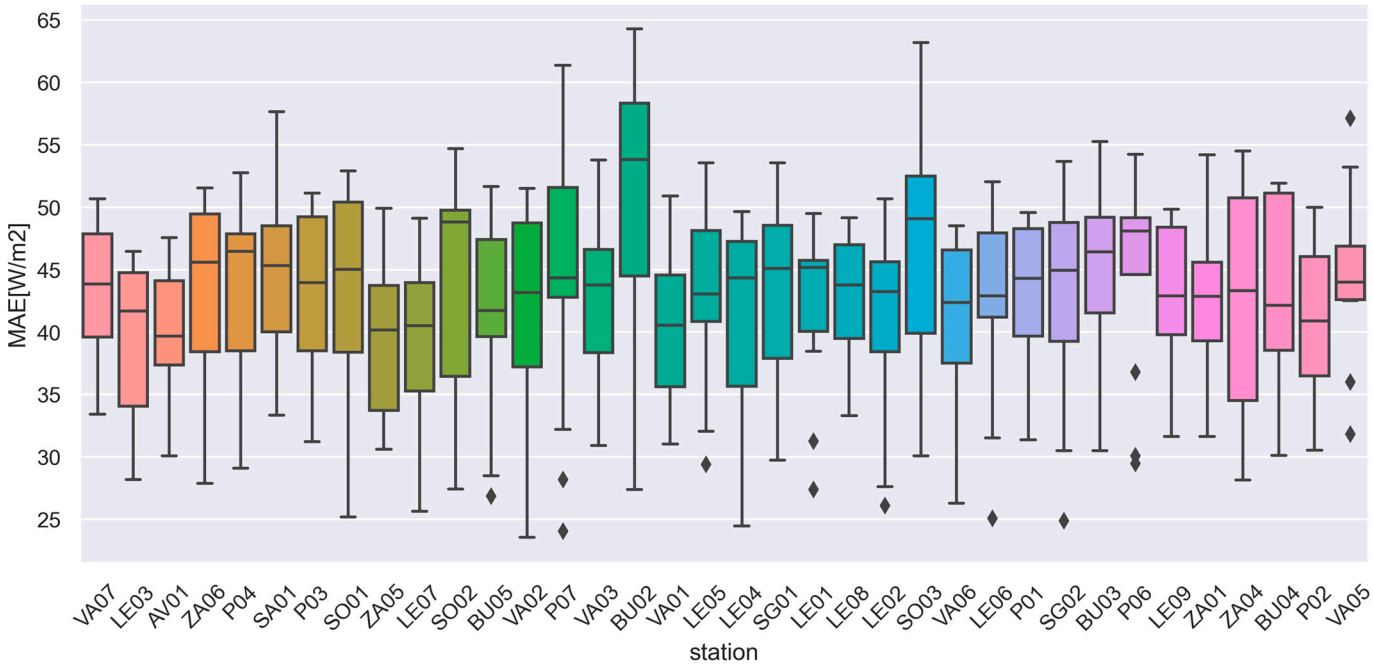


Figure 6. MAE distribution for each of the stations, where symbol ♦ represents extreme values.

Figure 7 presents a frequency plot comparing the values of the different data masks, dividing the results between known and unknown data. Each bar in the graph represents the frequency of occurrence of the observed values in each mask, allowing an analysis of how the predictions are distributed according to the amount of information available. The mask number represents its position within the chain. For each mask, the known data tend to be concentrated at higher frequencies, indicating that the model is more accurate when it has access to complete information. In contrast, the unknown data show a higher dispersion or lower frequencies, indicating that the model has more difficulty predicting when there is missing information. Although there are notable differences between known and unknown data, it is important to note that no matter the amount of unknown data, the model is always able to predict. This demonstrates that the model is robust even with unknown data.

Figure 8 shows the distribution of the R2 score for the different models and seasons. The R2 score, which measures the proportion of variability explained by the model, tends to be concentrated between 0.8 and 0.9 in most cases, indicating good overall performance. However, some important variations are observed: certain stations and models have lower R2 scores, even reaching values close to zero, indicating that in these situations the model is not able to learn the variability of the data. In general, the trend of the distribution is correct because as the R2 score increases, the MAE decreases. Analyzing the scattered cases, they are associated with stations where the data to be imputed exceed 60 percent of the sequence on many days.

Differences in the R2 score distribution between masks on known sequence and masks with an unknown part of the sequence reveal that models tend to perform more poorly when information is limited. In summary, although most of the results show solid performance, there are certain areas where model performance could be improved, especially in conditions with unknown data.

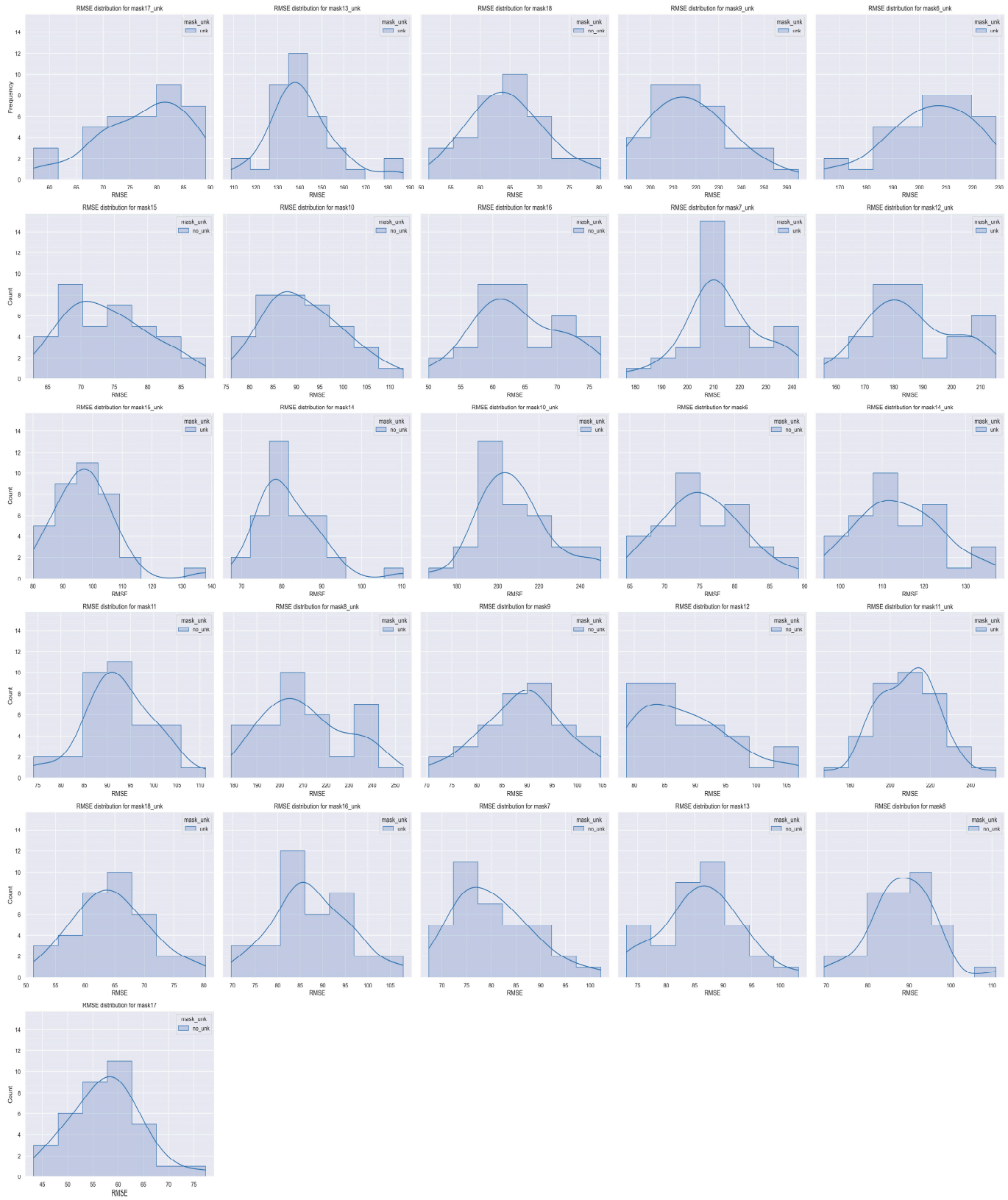


Figure 7. Frequency graph comparing the values of the different data masks, dividing the results between sequences with known and unknown data. In the title is the position of the mask, and if it is working with unknown data, the word 'unk'. Each bar in the graph represents the frequency of occurrence of the observed values in each mask.

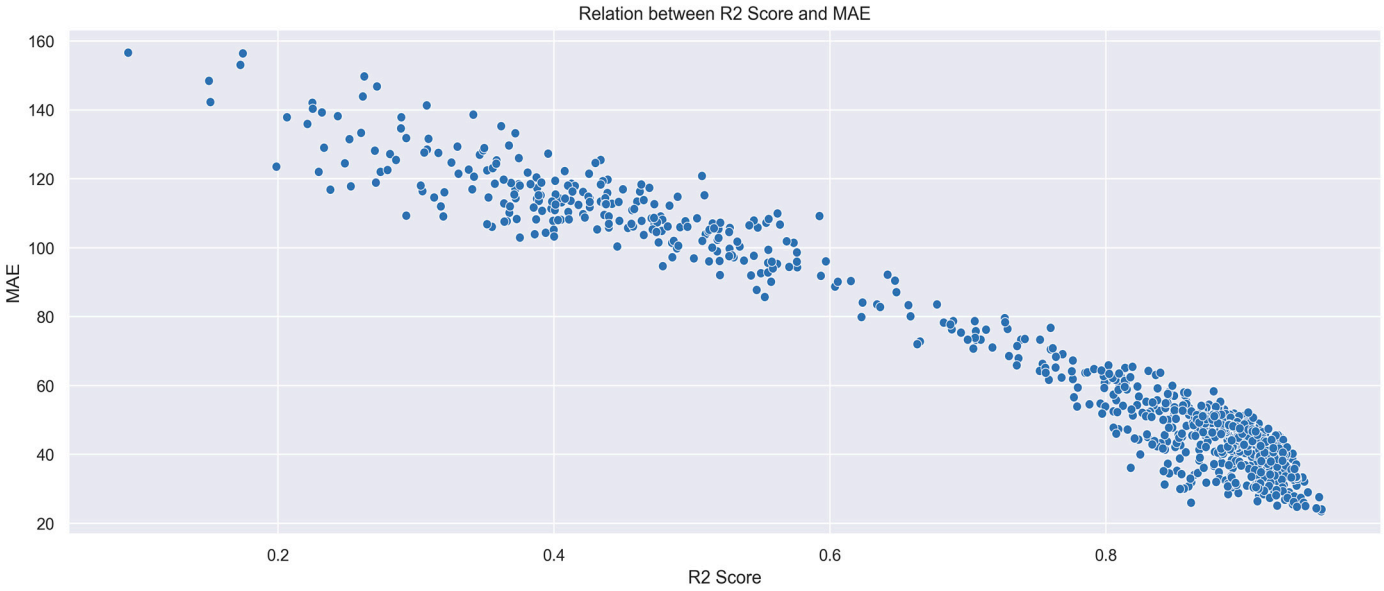


Figure 8. Relationship between R2 score and MAE for all stations with trained BERT.

4. Implementation of Phase_3 of the Proposed Methodology on a Dataset Established in the Literature from Hawaii, USA

4.1. Data: Hawaii Oahu Solar Measurement Grid

The data used corresponds to the dataset Hawaii Oahu Solar Measurement Grid [39], in particular, stations dh3 and ap7. They were selected based on their position within the grid with respect to the wind direction. Station dh3 benefits openly from the information of the neighboring stations because it is downwind, while station ap7 does not present downwind stations, as shown in Figure 9. This dataset is published with a temporal resolution of 1 s; in this study, it has been resampled to 10 s. The resulting dataset has no missing information, so in this case, only Phase_3 of the methodology will be applied.

4.2. Results

These results were obtained following Phase_3 of the methodology proposed in Section 2 (Figure 2). The metrics MAE, RMSE, and FS were used for regression models to evaluate the trained models. Table 2 shows the result of evaluating the three models ST_CNN_v1, ST_CNN_v2, and ST_Dilated_CNN on data from stations dh3 and ap7 of the Hawaii dataset.

Table 2. Results of evaluating models ST_CNN_v1, ST_CNN_v2, and ST_Dilated_CNN at stations dh3 and ap7.

Station	**ST_CNN_v1**			**ST_CNN_v2**			**ST_Dilated_CNN**		
Metric / **Station**	**FS (%)**	**MAE (W/m^2)**	**RMSE (W/m^2)**	**FS (%)**	**MAE (W/m^2)**	**RMSE (W/m^2)**	**FS (%)**	**MAE (W/m^2)**	**RMSE (W/m^2)**
dh3	63.50	9.51	30.22	63.70	8.87	30.07	63.70	9.26	30.06
ap7	48.54	11.57	41.34	50.16	11.29	40.29	49.89	11.56	40.25

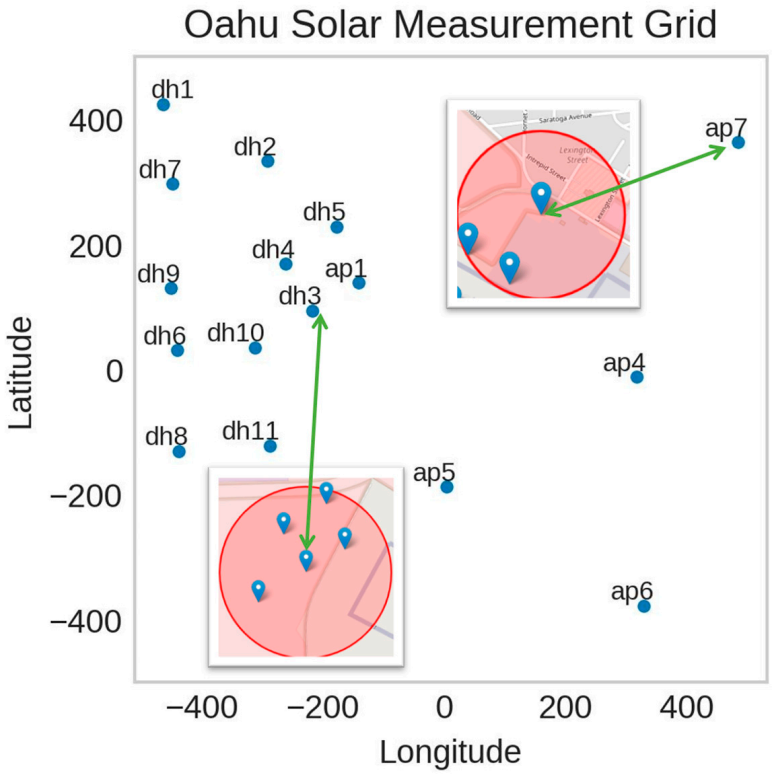


Figure 9. Spatial representation within the Oahu Solar Measurement Grid of the stations selected for the experiments (dh3 and ap7). dhn° identifies stations on the property of the Department of Hawaiian Homeland and apn° identifies stations on the property of Kalaeloa Airport.

For station dh3, the ST_Dilated_CNN and ST_CNN_v2 models show the best FS results, with a value of 63.70%. This value has a minimal difference with the result obtained by the ST_CNN_v1 model. The ST_CNN_v1 model has a difference of ±10 pp with respect to the other two models for the RMSE metric. As for FS, the ST_CNN_v2 and ST_Dilated_CNN models do not show significant differences for RMSE, unlike the MAE for station ap7, which has 3 pp above the ST_Dilated_CNN model with respect to the ST_CNN_v2 model.

The ST_CNN_v2 and ST_Dilated_CNN models have superior performance in all metrics with respect to the ST_CNN_v1 model. These results indicate that the models have the same behavior with these data as with the CyL-GHI data.

4.3. On the Variability of Model Performance

The same method of subsampling the data used in Ref. [30] was used to evaluate the models for this dataset. This method permits studying the consistency and variability of the models. For more detailed information on the implementation, see Ref. [30]. The mean and standard deviation of the proposed metrics were calculated on the results obtained after applying the subsampling method. Table 3 shows the result of this method for each of the models considering the two stations studied. The MAE is between 18.51 and 28.08, with a minimum deviation of 4.3 and a maximum of 7.73. The RMSE ranges between 50.75 and 84.49, with a minimum deviation of 9.25 and a maximum of 19.94. The R2 score ranges between 0.94 and 0.98, with a standard deviation of 0.01.

Table 3. Mean and standard deviation values of the performance metrics obtained.

Station	Model	MAE		RMSE		R2 Score	
		Mean	Std	Mean	Std	Mean	Std
ap7	ST_CNN_v1	28.08	7.73	84.49	19.94	0.94	0.02
	ST_CNN_v2	22.87	5.51	70.81	14.39	0.96	0.01
	ST_Dilated_CNN	25.1	6.27	69.9	13.98	0.96	0.01
dh3	ST_CNN_v1	21.77	5.49	55.5	11.28	0.97	0.01
	ST_CNN_v2	18.51	4.3	50.82	9.26	0.98	0.01
	ST_Dilated_CNN	19.76	4.68	50.75	9.25	0.98	0.01

For all of the models, the results obtained for station ap7 show the greatest differences, maintaining the same behavior as in other studies, which is expected because the amount of information it can receive from its neighbors is limited. While station dh3 achieves the best results just the opposite, being surrounded completely by stations as expected, its values are good for all models. These results establish the robustness of the models regardless of the dataset used.

5. Discussion

In this section, an analysis of the advantages and disadvantages of using the proposed methodology is presented.

The main advantage is that having an integrated workflow consisting of data acquisition, imputation, and prediction together with a description of the steps involved gives a clear overview and facilitates the training procedure. It should also be noted that the three proposed phases of the workflow can be applied separately, without depending on a specific model or step.

Another advantage is that using BERT for the imputation of missing data in solar irradiance time series can considerably improve the performance, as this transformer can learn the context, and has the ability to understand complex patterns in data and to generate training data with higher quality.

Moreover, this model offers a novel solution for the second scenario evaluated in the imputation, in which a specific position of the sequence is imputed, considering that all of the other values of the sequence from that position onwards are unknown. This scenario is the most interesting because when performing the imputation with a linear interpolation, if from one point the remaining values are unknown, the interpolation cannot continue beyond the last known value. In addition, the inclusion of DL models enables the ability to work with large volumes of information (such as the datasets used in this study).

Nonetheless, an important limitation is that the quality of the results achieved by applying this methodology is dependent on the climatic region for which the data are available. This problem has been established by other authors [40,41], hence the need for further evaluation of the methodology with additional data in additional scenarios to study its performance.

Regarding the dependence on the quality and completeness of the data, there are still important limitations in the generalizability of results due to dependence on the specific data used. For this reason, the participants involved should find and use high-quality data, updated and produced by trustworthy sources that declare the calibration of the instruments used for collecting data.

Another limitation of this study that can be mentioned is the use of a reduced number of datasets. It is recommended to explore the effectiveness of the proposed methodology in different contexts and use more datasets to further validate it.

6. Conclusions

In this work, a methodology for solar forecasting is presented. The methodology consists of a workflow that starts with data collection and ends with the forecasting step. The method is divided into three phases: Phase_1, related to the acquisition and preparation of the dataset, Phase_2, related to the imputation with a new proposed model, and Phase_3, related to the prediction with state-of-the-art deep learning models. The methodology has been applied to two publicly accessible datasets that are different in terms of geographic location and temporal resolution.

The first implementation of the methodology (on the CyL-GHI dataset) starts with the data collection step from Phase_1, while the second implementation (on the Hawaii Oahu Solar Measurement Grid Dataset, often used in the specialized literature as reference) starts with Phase_3, as it contains no missing data. The analysis of the results achieved proved the feasibility of the proposed methodology. In this regard, the application of Phase_1 enabled the creation of a new public dataset, CyL-GHI.

The main novelty is that, for Phase_2, a BERT model was trained from scratch for missing data imputation, which to the best of the authors' knowledge, is the first time a transformer model is pre-trained for irradiance data. In the case of the CyL-GHI dataset, this new imputation method for solar radiation based on transformers allowed an increase in performance of up to 3 percentage points (3%) compared to the traditional method. However, it must always be considered that in this field of solar prediction, the models are very closely associated with the data, so it is important to retrain the imputation model every time the location and data are changed.

The application of Phase_3 is based on selecting the state-of-the-art deep learning models for the regression task (in this case, based on convolutional neural networks). By separating the methodology into three phases, the method is flexible, allowing the application of different steps depending on the task to be carried out and on the available data.

As future work, it is proposed to evaluate the imputation model with new datasets. Fixed positions of the input string features referring to time and geographic location can also be introduced to evaluate whether these changes can help with better imputation. Another line of work could be to analyze the generalization capacity of the model when trained with data from different geographical locations.

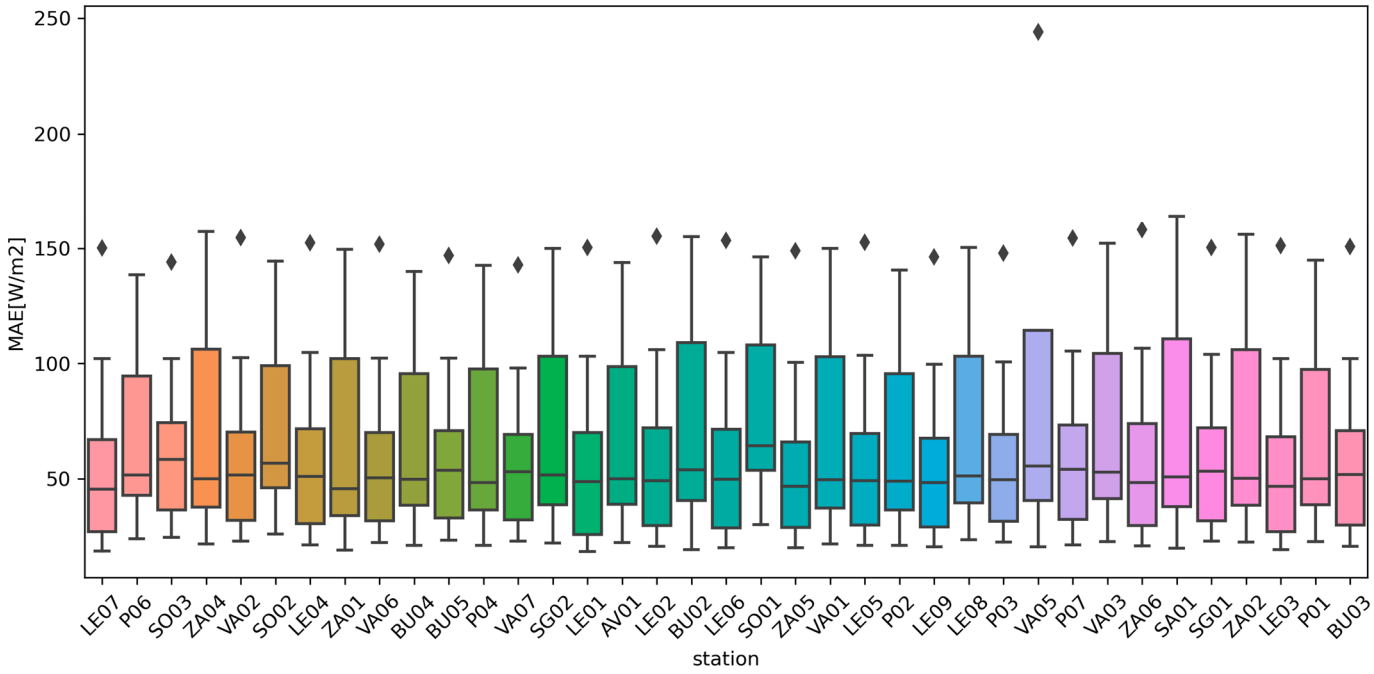
Author Contributions: Conceptualization, L.B.-C., M.-Á.M.-C. and C.-I.C.; methodology, L.B.-C. and M.-Á.M.-C.; software, L.B.-C.; validation, L.B.-C., M.-Á.M.-C. and C.-I.C.; investigation, L.B.-C.; resources, M.-Á.M.-C. and C.-I.C.; data curation, L.B.-C.; writing—original draft preparation, L.B.-C.; writing—review and editing, L.B.-C., M.-Á.M.-C. and C.-I.C.; visualization, L.B.-C.; supervision, M.-Á.M.-C. and C.-I.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

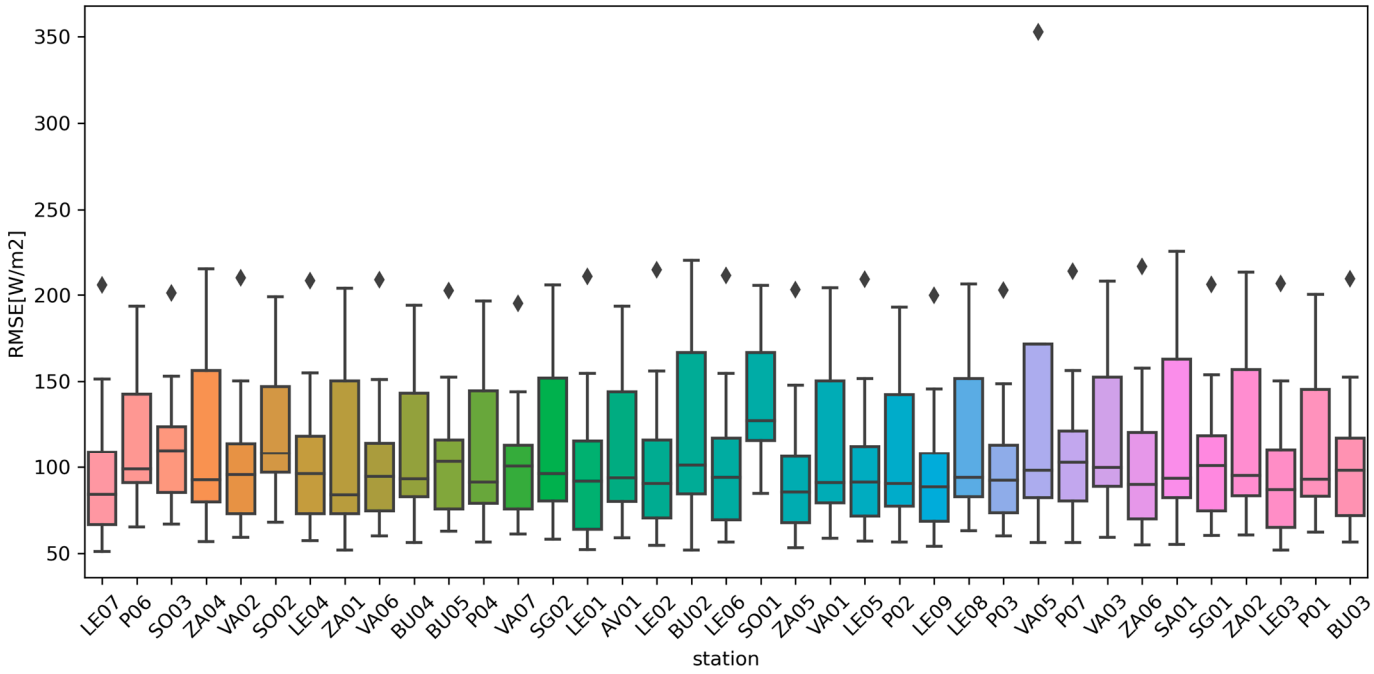
Data Availability Statement: The CyL-GHI dataset used in this study is openly available in the Zenodo repository (https://doi.org/10.5281/zenodo.7404167) (accessed on 6 June 2024). The second dataset is a public dataset provided by the National Renewable Energy Laboratory. The data are freely available at http://www.nrel.gov/midc/oahu_archive/ (accessed on 15 June 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. MAE Distribution for Each of the Stations of the CyL-GHI Dataset for the Linear Interpolation Model, Where Symbol ♦ Represents Extreme Values



Appendix B. RMSE Distribution for Each of the Stations of the CyL-GHI Dataset for the Linear Interpolation Model, Where Symbol ♦ Represents Extreme Values



References

1. Taye, M.M. Understanding of Machine Learning with Deep Learning: Architectures, Workflow, Applications and Future Directions. *Computers* **2023**, *12*, 91. [CrossRef]
2. Benavides Cesar, L.; Manso Callejo, M.Á.; Cira, C.-I.; Alcarria, R. CyL-GHI: Global Horizontal Irradiance Dataset Containing 18 Years of Refined Data at 30-Min Granularity from 37 Stations Located in Castile and León (Spain). *Data* **2023**, *8*, 65. [CrossRef]
3. Gandhi, O.; Zhang, W.; Kumar, D.S.; Rodríguez-Gallegos, C.D.; Yagli, G.M.; Yang, D.; Reindl, T.; Srinivasan, D. The value of solar forecasts and the cost of their errors: A review. *Renew. Sustain. Energy Rev.* **2024**, *189*, 113915. [CrossRef]
4. Diagne, M.; David, M.; Lauret, P.; Boland, J.; Schmutz, N. Review of solar irradiance forecasting methods and a proposition for small-scale insular grids. *Renew. Sustain. Energy Rev.* **2013**, *27*, 65–76. [CrossRef]
5. AlKandari, M.; Ahmad, I. Solar power generation forecasting using ensemble approach based on deep learning and statistical methods. *Appl. Comput. Inform.* **2019**, *20*, 231–250. [CrossRef]
6. Qing, X.; Niu, Y. Hourly day-ahead solar irradiance prediction using weather forecasts by LSTM. *Energy* **2018**, *148*, 461–468. [CrossRef]
7. Park, J.; Moon, J.; Jung, S.; Hwang, E. Multistep-ahead solar radiation forecasting scheme based on the light gradient boosting machine: A case study of Jeju Island. *Remote Sens.* **2020**, *12*, 2271. [CrossRef]
8. Ziyabari, S.; Du, L.; Biswas, S.K. Multibranch Attentive Gated ResNet for Short-Term Spatio-Temporal Solar Irradiance Forecasting. *IEEE Trans. Ind. Appl.* **2022**, *58*, 28–38. [CrossRef]
9. Dairi, A.; Harrou, F.; Sun, Y.; Khadraoui, S. Short-term forecasting of photovoltaic solar power production using variational auto-encoder driven deep learning approach. *Appl. Sci.* **2020**, *10*, 8400. [CrossRef]
10. Amaro e Silva, R.; Benavides Cesar, L.; Manso Callejo, M.Á.; Cira, C.-I. Impact of Stationarizing Solar Inputs on Very-Short-Term Spatio-Temporal Global Horizontal Irradiance (GHI) Forecasting. *Energies* **2024**, *17*, 3527. [CrossRef]
11. Chatfield, C. *The Analysis of Time Series*, 6th ed.; Chapman and Hall/CRC: New York, NY, USA, 2003. [CrossRef]
12. Fang, C.; Wang, C. Time Series Data Imputation: A Survey on Deep Learning Approaches. *arXiv* **2020**, arXiv:2011.11347. [CrossRef]
13. Glasbey, C.A. Imputation of missing values in spatio-temporal solar radiation data. *Environmetrics* **1995**, *6*, 363–371. [CrossRef]
14. Layanun, V.; Suksamosorn, S.; Songsiri, J. Missing-data imputation for solar irradiance forecasting in Thailand. In Proceedings of the 2017 56th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), Kanazawa, Japan, 19–22 September 2017; pp. 1234–1239. [CrossRef]
15. Demirhan, H.; Renwick, Z. Missing value imputation for short to mid-term horizontal solar irradiance data. *Appl. Energy* **2018**, *225*, 998–1012. [CrossRef]
16. Zhang, W.; Luo, Y.; Zhang, Y.; Srinivasan, D. SolarGAN: Multivariate Solar Data Imputation Using Generative Adversarial Network. *IEEE Trans. Sustain. Energy* **2021**, *12*, 743–746. [CrossRef]
17. De-Paz-Centeno, I.; García-Ordás, M.T.; García-Olalla, Ó.; Alaiz-Moretón, H. Imputation of missing measurements in PV production data within constrained environments. *Expert Syst. Appl.* **2023**, *217*, 119510. [CrossRef]
18. Yldz, A.Y.; Koc, E.; Koc, A. Multivariate Time Series Imputation with Transformers. *IEEE Signal Process. Lett.* **2022**, *29*, 2517–2521. [CrossRef]
19. Bansal, P.; Deshpande, P.; Sarawagi, S. Missing value imputation on multidimensional time series. *Proc. VLDB Endow.* **2021**, *14*, 2533–2545. [CrossRef]
20. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *2017*, 5999–6009.
21. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv:1810.04805. [CrossRef]
22. Koroteev, M.V. BERT: A Review of Applications in Natural Language Processing and Understanding. *arXiv* **2021**, arXiv:2103.11943. [CrossRef]
23. Dong, X.; Bao, J.; Zhang, T.; Chen, D.; Zhang, W.; Yuan, L.; Chen, D.; Wen, F.; Yu, N. Bootstrapped Masked Autoencoders for Vision BERT Pretraining. In Proceedings of the 17th European Conference on Computer Vision (ECCV), Tel Aviv, Israel, 23–27 October 2022; pp. 247–264. [CrossRef]
24. Wang, R.; Chen, D.; Wu, Z.; Chen, Y.; Dai, X.; Liu, M.; Jiang, Y.-G.; Zhou, L.; Yuan, L. BEVT: BERT Pretraining of Video Transformers. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 14713–14723. [CrossRef]
25. Lee, H.; Lee, S.; Lee, I.; Nam, H. AMP-BERT: Prediction of antimicrobial peptide function based on a BERT model. *Protein Sci.* **2023**, *32*, e4529. [CrossRef] [PubMed]
26. Ghazikhani, H.; Butler, G. TooT-BERT-M: Discriminating Membrane Proteins from Non-Membrane Proteins using a BERT Representation of Protein Primary Sequences. In Proceedings of the 2022 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), Ottawa, ON, Canada, 15–17 August 2022; pp. 1–8. [CrossRef]

27. Wen, N.; Liu, G.; Zhang, J.; Zhang, R.; Fu, Y.; Han, X. A fingerprints based molecular property prediction method using the BERT model. *J. Cheminform.* **2022**, *14*, 71. [CrossRef] [PubMed]
28. Gao, Y.; Xiong, Y.; Wang, S.; Wang, H. GeoBERT: Pre-Training Geospatial Representation Learning on Point-of-Interest. *Appl. Sci.* **2022**, *12*, 12942. [CrossRef]
29. Cesar, L.B.; Manso-Callejo, M.-Á.; Cira, C.-I. BERT (Bidirectional Encoder Representations from Transformers) for Missing Data Imputation in Solar Irradiance Time Series. *Eng. Proc.* **2023**, *39*, 26. [CrossRef]
30. Benavides Cesar, L.; Manso-Callejo, M.-Á.; Cira, C.-I. Three Novel Artificial Neural Network Architectures Based on Convolutional Neural Networks for the Spatio-Temporal Processing of Solar Forecasting Data. *Appl. Sci.* **2024**, *14*, 5955. [CrossRef]
31. Journée, M.; Bertrand, C. Quality control of solar radiation data within the RMIB solar measurements network. *Sol. Energy* **2011**, *85*, 72–86. [CrossRef]
32. Van Rossum, G.; Drake, F.L. *Python 3 Reference Manual*; CreateSpace: Scotts Valley, CA, USA, 2009.
33. Chollet, F.; Gardener, T.; Zhu, Q.S.; Jin, H.; Rahman, F.; Watson, M.; Chiu, H.; Qian, C.; Lee, T.; de Marmiesse, G.; et al. "Keras". GitHub. 2015. Available online: https://github.com/fchollet/keras (accessed on 6 June 2024).
34. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: A system for large-scale machine learning. In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), Savannah, GA, USA, 2–4 November 2016; pp. 265–283.
35. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Müller, A.; Nothman, J.; Louppe, G.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830. Available online: http://arxiv.org/abs/1201.0490 (accessed on 6 June 2024).
36. Van Der Walt, S.; Colbert, S.C.; Varoquaux, G. The NumPy array: A structure for efficient numerical computation. *Comput. Sci. Eng.* **2011**, *13*, 22–30. [CrossRef]
37. Waskom, M.; Botvinnik, O.; O'Kane, D.; Hobson, P.; Lukauskas, S.; Gemperline, D.C.; Augspurger, T.; Halchenko, Y.; Cole, J.B.; Warmenhoven, J.; et al. mwaskom/seaborn: v0.8.1 (September 2017) (v0.8.1). Zenodo. 2017. Available online: https://doi.org/10.5281/zenodo.883859 (accessed on 13 January 2025).
38. Hunter, J.D. Matplotlib: A 2D Graphics Environment. *Comput. Sci. Eng.* **2007**, *9*, 90–95. [CrossRef]
39. Segupta, M.; Andreas, A. *Oahu Solar Measurement Grid (1-Year Archive): 1-Second Solar Irradiance*; Data: Oahu, HI, USA, 2010. [CrossRef]
40. Yang, D.; Wang, W.; Gueymard, C.A.; Hong, T.; Kleissl, J.; Huang, J.; Perez, M.J.; Perez, R.; Bright, J.M.; Xia, X.; et al. A review of solar forecasting, its dependence on atmospheric sciences and implications for grid integration: Towards carbon neutrality. *Renew. Sustain. Energy Rev.* **2022**, *161*, 112348. [CrossRef]
41. Cirés, E.; Marcos, J.; de la Parra, I.; García, M.; Marroyo, L. The potential of forecasting in reducing the LCOE in PV plants under ramp-rate restrictions. *Energy* **2019**, *188*, 116053. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.