IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 20 June 2025, accepted 4 July 2025, date of publication 15 July 2025, date of current version 21 July 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3589131

RESEARCH ARTICLE

An Optimized Hybrid Framework Based on Long-Short Term Memory Neural Networks and Fourier SynchroSqueezed Transform for Photovoltaic Power Forecasting

SAMER RAJAH, FRANCISCO J. MUÑOZ, AND ALEJANDRO RODRÍGUEZ
Electrical Engineering Department, University of Málaga, 29071 Málaga, Spain

Corresponding author: Alejandro Rodríguez (arodriguezg@uma.es)

This work was supported in part by the Ministry of Science and Innovation of Spain through the Intelligent Grid-Forming EV Hubs: Power Processing, Control, Operation and Exploitation, within the framework of the Knowledge Generation Projects call under Project PID2022-142372OB-C21; in part by European Union through the European Regional Development Fund (ERDF); and in part by Universidad de Málaga / CBUA for open access charge.

ABSTRACT Accurate prediction of photovoltaic power is crucial to optimize its integration into the power grid. However, this task is highly complex due to the inherently stochastic nature of photovoltaic power generation. To address this challenge, this paper proposes a novel hybrid framework that combines a Long Short-Term Memory network with the Fourier SynchroSqueezed Transform, and Bayesian Optimization. The model integrates the Fourier SynchroSqueezed Transform with the Long Short-Term Memory network to enhance the identification of very short-term patterns in photovoltaic power production. Additionally, Bayesian Optimization is used to determine the most relevant hyperparameters of the Long Short-Term Memory network. The model was evaluated using real-world data from the Desert Knowledge Australian Solar Centre project, considering different data segmentation sizes, ranging from 15 days to four months, and input types, including univariate and multivariate data, with prediction horizons ranging from five minutes to three days. Significant improvements in prediction accuracy were observed. For example, the Root Mean Squared Error for the 15-day data segmentation decreased by 19.48% when using multivariate inputs and by 29.59% for univariate inputs. For a four-month data segmentation, improvements reached 40.12% and 64.47%, respectively. Furthermore, the model demonstrated robust performance across various prediction horizons. For the 15-day segmentation, the Root Mean Squared Error was approximately 0.707 kW with an average power of 8.540 kW, while for the four-month segmentation, the error ranged around 0.467 kW with an average power of 4.733 kW. These results demonstrate the effectiveness and consistency of the proposed model in enhancing photovoltaic power forecasts across different time scales and data segmentations. The demonstrated ability to provide consistent and accurate predictions across multiple time horizons and data segmentations represents a significant advancement toward seamless integration of solar energy into the electricity grid.

INDEX TERMS Deep learning, long short-term memory neural network, Fourier SynchroSqueezed transform, Bayesian optimization, photovoltaic power prediction.

I. INTRODUCTION

The generation of energy from photovoltaic sources has garnered considerable attention worldwide, driven by initiatives aimed at neutrality and the reduction of carbon emissions. Photovoltaic energy, with its inherent advantages, is experiencing noteworthy each year. However, the output power of photovoltaic systems is heavily influenced by meteorological variables [1] such as solar irradiation, temperature, cloud cover, wind speed, etc. This set of

The associate editor coordinating the review of this manuscript and approving it for publication was Ahmed F. Zobaa.

© 2025 The Authors. This work is licensed under a Creative Commons Attribution 4.0 License.
For more information, see https://creativecommons.org/licenses/by/4.0/

factors results in highly variable energy that changes over time.

This variability poses challenges to the efficient operation of the electrical grid, especially when electrical operators are planning how to distribute that variable energy across the grid to match real-time generation and consumption. Therefore, accurate prediction of photovoltaic energy facilitates the management and integration of this energy into the grid, as it allows the electrical operator to anticipate the amount of energy that will be injected in a specific period for network performance analysis, reducing uncertainty in predictions and enhancing the integration of photovoltaic energy into the grid.

A. RELATED WORKS

Predictive models for photovoltaic energy can be classified based on the prediction horizon or the prediction model used [2], although there is no unanimous consensus in defining this classification. According to the prediction horizon, photovoltaic energy can be categorised [3] into intra-hour, intra-day, and day-ahead predictions. In intra-hour predictions, the prediction horizon covers from a few minutes to one hour in advance, while in intra-day predictions, the prediction horizon is limited from one hour to six hours. Finally, in day-ahead predictions, the prediction horizon is limited from 6 hours to 72 hours, usually with an hourly resolution. All these horizons are related to the activities of the electrical grid operator, such as network stability, energy dispatch, balancing production and consumption in the electricity market, variability related to operations, optimization, and planning. Depending on the model used, prediction models can be classified into physical models, statistical models, machine learning models, and their hybrid combinations.

Physical atmospheric models are based on the interaction between the physical state and the dynamic movement of solar radiation in the atmosphere. These models are further categorised into Numerical Weather Prediction (NWP), sky imagery models, and satellite imagery. NWP is accurate for daily horizons and provides predictions up to 15 days in advance. Sky and satellite imagery models are accurate for intra-day horizons [4]. Physical models are accurate for predicting photovoltaic energy for daily and intra-day horizons. However, they require expensive equipment and parameters to produce meteorological data [5].

Statistical models are another commonly used technique in time series prediction. These models use mathematical equations to extract patterns and correlations from historical data. These models are easy to implement as they do not require detailed information about system's internal states for modelling. However, they have some limitations regarding adaptability, learning capability, and prediction accuracy [6]. These techniques accurately predict photovoltaic energy for the intra-hour prediction horizon. However, as the prediction time increases, they may lose precision. The main disadvantage of the statistical models is the lack of physical modelling during the prediction 'the nonlinearity.'

To address the non-linearity problem, many authors have utilised machine learning models [5], [6], [7], [8], including Artificial Neural Networks (ANNs), K-Nearest Neighbor (KNN), Support Vector Machines (SVM), Random Forest (RF), among others. These models do not require a mathematical model to learn from historical data and analyse the stochastic behaviour between the past and the future [9]. However, one of the main disadvantages of shallow Machine Learning (ML) models is their relative lack of depth. Shallow models exhibit notable limitations, such as the learning process, parameter selection, and low generative capacity [10]. Consequently, these models may be unsuitable for learning the complex patterns in photovoltaic power data. Furthermore, the complexity of these models becomes an obstacle as the data size increases. While shallow models perform well when the training set is relatively small, an increase in data can lead to instability and lack of convergence due to the abundance of photovoltaic power data. All these limitations have led to the consideration of using Deep Learning (DL) techniques.

Recently, with advancements in supercomputing and the availability of large datasets, DL, as a promising branch of ML, has piqued the interest of many researchers in utilising these techniques to enhance accuracy in predicting photovoltaic energy generation. These techniques possess a high generative capacity, unsupervised learning ability, and can handle large datasets, thus overcoming limitations in shallow models. DL techniques have been successfully applied in various areas, including forecasting of power generation [11], [12], [13].

Several DL architectures have been employed for photovoltaic energy prediction [10], including: Deep Belief Networks (DBN), Stacked Autoencoders (SAE), Deep Recurrent Neural Networks (RNN), and other architectures such as Stacked Extreme Learning Machines (SELM), Deep Reinforcement Learning, Deep Convolutional Neural Networks (DCNN), and Generative Adversarial Networks (GAN), among others.

The first application of DL techniques in the field of renewable energy power prediction was introduced by [14], where the authors utilised DBN, SAE and Long Short-Term Memory (LSTM) in 21 solar power plants. The results show that these techniques yield better outcomes compared to the Multilayer Perceptron (MLP) and physical models. One of the DL techniques is CNNs, a specialised type of neural network designed to process grid-like data, such as images, and perform well in time series prediction and pattern recognition. Consequently, CNNs can understand the relationship between sky images and photovoltaic power. An outstanding example is the SolarNet model developed in [15], where the authors use a CNN to predict solar energy 10 minutes in advance. This model, which only took sky images as input without numerical measurements, demonstrates the ability to learn latent patterns between images and global horizontal irradiance.

Time-frequency transform (TFTs) have been used in combination with DL techniques. For instance, Variational Mode Decomposition (VMD) was employed alongside Recurrent Neural Network, with their hyperparameters optimized using an original modified version of the Harris Hawk Optimization algorithm for renewable power prediction, including both wind and photovoltaic sources [16]. Among DL architectures, LSTM in particular has also been widely applied in conjunction with these TFTs-for example, with the Fourier transform as a feature extractor [17] and as a denoising method [18]. Notable results have also been achieved by combining LSTM with VMD [19] and with wavelet transform [20], [21].

LSTM has also demonstrated its superiority over other techniques such as shallow Machine Learning methods and other deep learning-based techniques in photovoltaic energy prediction [10], [22], [23], [24], and wind power prediction [25].

Different configurations of an LSTM network to predict photovoltaic energy are considered in [26], where a configuration of three inputs (3-time steps) and 50 epochs provides good results. In addition, the comparison with other shallow models, such as Multiple Linear Regression (MLR), Neural Network (NN), showed that the LSTM model performs better. In the same line [27] develops different types of deep learning neural networks, including LSTM, GRU, BiLSTM, BiGRU, $CNN - LSTM_{1D}$ and $CNN_{1D} - GRU$, to predict photovoltaic energy at different prediction horizons (1 min, 5 min, 30 min and 60 min). The best result was obtained using LSTM for a 1-min prediction horizon, but as the horizon increased, the error also increased. The hyperparameter determination was exhaustive but time-consuming. Also, in [10] the authors developed an LSTM for a day-ahead PV power forecasting, demonstrating the superiority of such an LSTM-based approach with respect to Back Propagation (BP), Multi-Layer Perceptron (MLP), Radial Basis Function (RBF), Fuzzy C-Mean Clustering, Layer Recurrent (LR), Autoregressive Moving Average (ARMA), Autoregressive Integrated Moving Average (ARIMA, Seasonal Autoregressive Integrated Moving Average (SARIMA), and persistent model [22], [23].

Intending to improve prediction accuracy, other authors focused on the development of hybrid models. For example, in [28] three deep neural network techniques- CNN, LSTM, and a hybrid CNN-LSTM model- were utilised to predict photovoltaic energy. Their results show, on the one hand, that the accuracy of the results heavily depends on the size of the available data, and on the other hand, the development hybrid model had higher accuracy than individual CNN and LSTM models. A constrained LSTM (C-LSTM) [29] is proposed by using KNN to extract prior knowledge to combine with transfer learning. These works present effective results but are limited to a given prediction horizon.

There are works that overcome this limitation, [30] where LSTM is combined with a non-linear autoregressive Neural Network optimized by the Tabu search algorithm considers two prediction horizons: 5 minutes and 1 hour. Also Boosting techniques hybridising with LSTM [31] worked with various prediction horizons (from 5 minutes to 14 days). However, the results show that the model's performance improves as the time horizon decreases. A novel and effective hybridization of the Particle Swarm Optimization (PSO) algorithm was introduced to optimize the hyperparameters of a LSTM for power PV prediction demonstrating promising results in improving model performance [32].

B. CONTRIBUTIONS

The procedure described above provides well-established photovoltaic generation prediction models based on machine learning techniques, which typically entail high computational costs. These learning processes often require the optimization of a large number of hyperparameters, further increasing computational demands [33]. Moreover, existing models generally achieve satisfactory performance by targeting specific prediction horizons and relying on fixed data set segmentation (DSS).

This work introduces a novel model capable of delivering accurate predictions across varying time horizons independently of the DSS. The proposed model consists of three core parts. First, the Fourier Synchrosqueezed Transform (FSST) is employed to enhance adaptation to temporal signal variations, offering a more precise representation of frequency components while preserving the original signal's temporal resolution. Second, a LSTM network is used, whose architecture enables effective control of input flow and internal memory, thus improving the model's ability to capture temporal dependencies. Third, Bayesian Optimization (BO) is applied to tune the LSTM's hyperparameters efficiently, ensuring improved performance while maintaining manageable computational costs.The innovative integration of FSST with a Bayesian-optimized LSTM network results in a photovoltaic generation prediction model that performs robustly across multiple prediction horizons and DSS configurations, striking a favorable balance between accuracy and computational efficiency.

The main contributions of this work are summarized as follows:

- The use of the FSST enables the identification of very short-term patterns in time series data. This appears to be the first application of FSST in the context of photovoltaic energy production forecasting, as far as can be determined from the existing literature.
- The proposed integration of the FSST with a hyperparameter-optimized LSTM network exhibits strong predictive performance across a wide range of time horizons (5 minutes, 30 minutes, 1 hour, 6 hours, 1 day, and 3 days) and data set segmentation (DSS), ranging from 15 days to 4 months.

To ensure consistency and robustness of the proposed hybrid approach, the following methodological considerations were addressed:

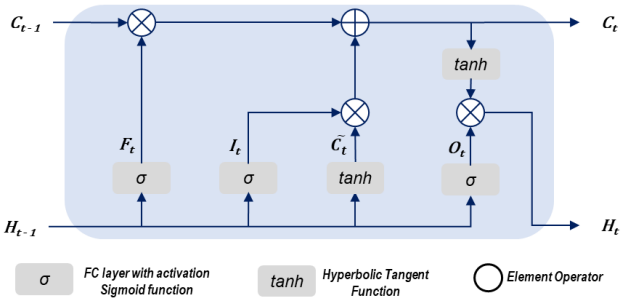


FIGURE 1. The internal structure of an LSTM cell.

- Two types of input patterns: (i) Univariate input, where only the historical evolution of total active power is used to forecast future power output; and (ii) Multivariate input, where three meteorological variables—global horizontal irradiance (W/m^2), diffuse solar radiation (W/m^2) and ambient temperature (^oC)—are used as predictors of future active power output (kW).
- Hyperparameter optimization: BO is applied to determine the optimal configuration of LSTM hyperparameters, taking into account the input type, DSS, and prediction horizon.
- Evaluation criteria: The performance of the proposed model is evaluated based on both prediction accuracy and computational cost.

C. ORGANIZATION

The remainder of this paper is organised as follows: Materials and method are presented in Section II, which includes an overview of LSTM, the integration of FSST in the data preparation phase of the LSTM network, and the search for optimal hyperparameters through BO, along with the evaluation metrics. Section III details the data, its characteristics, and the cleaning, scaling, and feature selection processes. It also explains the case studies carried out. Finally, Section IV presents the main conclusions and future perspectives.

II. MATERIALS AND METHODS

A. LONG SHORT-TERM MEMORY (LSTM)

LSTM networks were originally proposed by Hochreiter and Schmidhuber in 1997 [34] as an advancement over conventional neural networks by introducing memory blocks in place of standard hidden units. This design addresses the vanishing gradient problem by enabling gradients to propagate through many time steps without vanishing or exploding [35]. In LSTM terminology, 'Long-Term Memory' denotes the network's capability to preserve relevant information over extended sequences via slowly adapting weight parameters. These parameters encode generalized knowledge from the training data, allowing the network to retain information from prior time steps for use in subsequent decision processes. Conversely, 'Short-Term Memory' refers to the temporary retention of information within a sequence, maintained as transient activations transmitted between successive nodes. The core LSTM architecture comprises a memory cell containing an internal state alongside a set of multiplicative gates, as depicted in Figure 1. These gates modulate the information flow by controlling which inputs are stored within the cell and the extent to which the cell state influences the network's output.

The data feeding the mentioned gates and the input node $\tilde{C}_t$ consists of the input at the current time step (X_t) and the hidden state from the previous time step (H_{t-1}). The input gate (I_t) determines how much new input data should be considered through $\tilde{C}_t$. The forget gate (F_t) decides what information to retain and what to discard from the previous internal state of the cell (C_{t-1}). These two gates give the LSTM model flexibility to learn when to keep an input unchanged and when to perturb it based on successive inputs. The output gate (O_t) determines whether the memory cell should influence the output at the current time step, which is represented by the current hidden state of the cell (H_t). When the output gate value is close to 1, the internal state of the memory cell affects the layers without obstacles. On the other hand, when the value is close to 0, the memory cell is prevented from affecting other layers of the network at the current time step. In other words, the memory cell can accumulate information over many time steps with or without affecting the rest of the network, depending on the value of the output gate. Mathematically, the equations representing the LSTM cell can be defined as follows:

$$F_t = \sigma(W_{xf}X_t + W_{hf}H_{(t-1)} + b_f) \quad (1)$$
$$I_t = \sigma(W_{xi}X_t + W_{hi}H_{(t-1)} + b_i) \quad (2)$$
$$O_t = \sigma(W_{xo}X_t + W_{ho}H_{(t-1)} + b_o) \quad (3)$$
$$\tilde{C}_t = tangh(W_{xc}X_t + W_{hc}H_{(t-1)} + b_c) \quad (4)$$
$$C_t = F_t \cdot C_{t-1} + I_t \cdot \tilde{C}_t \quad (5)$$
$$H_t = O_t \cdot tangh(C_t) \quad (6)$$

where W_{xf}, W_{xi}, W_{xo}, W_{xc} y W_{hf}, W_{hi}, W_{ho}, W_{hc} are weight parameters of the neurons, b_f, b_i, b_o, b_c are bias parameters, $\cdot$ is a product operator, and σ is a sigmoid function.

B. FOURIER SYNCHROSQUEEZED TRANSFORM

In this paper, the use of the FSST allows analysis of temporal patterns in time series. FSST can be instrumental in identifying frequency patterns in the data, a crucial aspect of understanding very short-term variations in photovoltaic energy production.

The FSST is based on the Short-Time Fourier Transform (STFT). STFT is a technique used to analyse how the frequencies of a signal vary over time. It is based on the fact that a signal is piecewise stationary over short time intervals. This means that the signal can be considered constant over a small interval, which allows the Fourier transform to be applied to analyse the frequencies in that interval. However, many signals are not stationary in the short term (speech signals, music, ...). This is the main limitation of this

technique. FSST is a spectral energy reallocation method in the Time-Frequency domain [36]. This technique allows a better adaptation to temporal signal variation and provides a more accurate representation of the frequencies present in non-stationary signals.

FSST takes the best of both techniques to accurately and efficiently represent non-stationary signals in the Time-Frequency domain [37]. The STFT is used to divide the signal into segments, which is helpful for capturing temporal variations. Meanwhile, the SST is applied to each segment to improve the accuracy of frequency identification. The FSST mathematically can be defined by the equation:

$$FSST = \int_{-\infty}^{\infty} v_f(\eta, t) \cdot \delta\Big(w - w_f(\eta, t)\Big) d\eta \qquad (7)$$

where $v_f(\eta, t)$ is the short-time Fourier Transform (STFT) of a signal at time t filtered at frequency η using a spectral window f, w y w_f are the observed frequency and instantaneous frequency, respectively. It is worth mentioning that the FSST maintains the same temporal resolution as the raw signal. Given this feature, it is expected that the FSST will improve the prediction results. The optimal size of the FSST depends on several factors, including the characteristics of the data, the level of detail to be captured in the data and others.

In comparison to other time-frequency analysis methods, FSST stands out due to its sharp and noise-robust representation, which is essential for characterizing non-stationary signals like photovoltaic energy output. Classical techniques such as the Short-Time Fourier Transform [38], Continuous Wavelet Transform (CWT) [39], or Wigner-Ville Distribution (WVD) [40] offer basic time-frequency analysis capabilities, but their non-parametric nature often results in trade-offs between time and frequency resolution or undesirable artifacts in the time-frequency representation (TFR). More advanced transforms, like the Smoothed Pseudo WVD [41] or Choi-Williams Distribution (CWD) [42], improve these limitations but still suffer from cross-term interference or limited adaptability.

SST techniques [43], including the Wavelet-based SST (WSST) and the Fourier-based SST (FSST), have emerged to address these drawbacks by reallocating spectral energy and enhancing resolution. WSST performs well for slowly-varying components [44], but struggles with fast-varying, multicomponent signals [45]. FSST improves upon this by reassigning energy based on STFT, providing sharper TFRs with improved resolution for both slow and moderately fast components [43], [46]. Furthermore, it retains the temporal resolution of the original signal, which is crucial in short-term forecasting applications.

C. HYPERPARAMETERS OPTIMIZATION

LSTM networks, like other deep learning models, have several hyperparameters. Some of these hyperparameters are related to computational capacity, while others affect the model's performance and its ability to make accurate predictions. The configuration of these hyperparameters impacts the overall accuracy of the model depending on the behaviour and the data size. Poorly configured hyperparameters can lead to problems of overfitting or underfitting during training [47].

The task of tuning the hyperparameters is not straightforward, can be time-consuming, and requires an understanding of how each hyperparameter affects the overall performance of the model. There are different classical strategies for this tuning, including trial and error, grid search and random search. These strategies, valid in certain cases, have different disadvantages: they are time-consuming, their efficiency decreases as the number of hyperparameters to be optimized increases, and they can be ineffective because they require many iterations to obtain satisfactory results [48]. Along this path, different metaheuristics have been applied for LSTM hyperparameters optimization, such as neural networks, genetic algorithm, particle swarm optimization [49], [50], [51], and original hybridizations of optimization techniques [25] and [32].

BO has been widely employed to enhance hyperparameters of different DL architectures. BO allows, smart and efficiently, to find the optimal hyperparameter settings with a minimum number of objective function evaluations [52]. This approach is especially valuable when objective function evaluations are costly (prediction) and access to the gradient of the loss function concerning the hyperparameters is not available. BO uses the information accumulated throughout the optimization to perform an efficient search.

BO comprises two key elements: a surrogate (probabilistic) model and an acquisition function. The surrogate model approximates the objective function being optimized; this model takes a probability distribution over the possible objective functions using the Gaussian Process (GP). A mean and covariance function fully specify the GP, which thus provides an estimate of the objective function along with a confidence interval. This means it estimates not only the objective function at a given point but also the uncertainty associated with that estimate. The quality of the GP is highly dependent on the covariance function (also called Kernel function) [53]. The acquisition function serves as a criterion for selecting the next point to evaluate the objective function, considering the probability distribution of the surrogate model. This function determines the utility of evaluating the objective function in different areas of the search space. In the present work, this approach involves selecting points that are highly likely to be suitable and can also be evaluated quickly since the search is oriented towards low-cost areas.

BO involves making decisions under uncertainty, trading off exploration and exploitation. As discussed previously, LSTM networks excel at leveraging past data to generate precise predictions in sequential contexts. The last piece of puzzle is to connect the LSTM network to BO, here we employ BO with Gaussian process surrogate model to efficiently guide the search for optimal LSTM hyperparameters.

As summarized in Algorithm 1, the procedure begins by defining a prior over the space of LSTM hyperparameters (number of hidden layers, number of neurons, learning rate

Algorithm 1 Hyperparameters Tuning Using the GP Model

```
Input data D = {X, Y}
Define the hyperparameter search space S
Randomly select an initial subset D_0 = {X_0, Y_0} from D
Fit a Probabilistic Surrogate Function to find Y_0 = G(X_0)
for t = 1, 2, ... do
    Optimize α ← X_t = arg max_x α(X; D_{t-1})
    Train LSTM using X_t and compute validation loss Y_t
    Augment data set: D_t = D_{t-1} ∪ (X_t, Y_t)
    Update the surrogate model G with new data D_t
end for
```

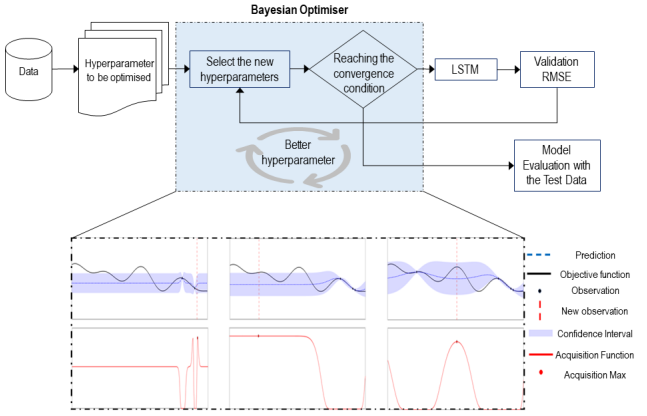


FIGURE 2. The internal structure of an LSTM cell.

and L2 regularization). An initial set of configurations is randomly sampled and evaluated by training the LSTM and recording its validation loss. A GP model is then fit to this initial dataset to serve as a probabilistic surrogate for the true validation loss function. At each iteration of BO, the acquisition function is optimized to propose a new hyperparameter configuration that maximizes the potential for improvement, given the uncertainty encoded by the GP model. This new configuration is then evaluated by training an LSTM model, and the resulting performance is used to update the surrogate model. This process is repeated iteratively gradually refining the model's understanding of the objective landscape.

In this paper, BO is employed to find the optimal values of different LSTM hyperparameters. Mathematically, x^* of the LSTM network that satisfies Equation 8.

$$x^* = arg\ min_{x \in X} f(x) \tag{8}$$

where x are the candidates hyperparameters to optimize and $f(x)$ is the objective function to minimise. The objective function will be one of the metrics that allows the quality of the photovoltaic power prediction to be evaluated compared to the actual measured value.

Figure 2 illustrates the BO procedure in three iterations where the objective function at the bottom is only a visual example of BO procedure, which is unknown in practice. In this example, the operation of BO on a one-dimensional 1-D function is illustrated. The objective is to minimise the distance between the dashed line and the solid line, using the probabilistic surrogate model of the Gaussian process (the predictions are shown as a dashed blue line, with a blue tube representing the confidence interval), by maximising the acquisition function defined by the lower red curve.

TABLE 1. Hyperparameters to be optimized.

Hyperparameters	Search space
Number of hidden layers	[1 3]
Number of neurons	[75 200]
Learning rate	[10^{-2} 1]
L2 Regularization	[10^{-10} 10^{-2}]

In automatic hyperparameter optimization, different hyperparameters can be considered. The first set is related to the design of the model. In all DL models, there is an input, hidden, and output layer. The complexity of a model depends on the number of hidden layers and the number of neurons in each hidden layer. Therefore, these two hyperparameters are chosen as hyperparameters to be optimized. The optimization of the number of neurons was implemented for each hidden layer.

Other hyperparameters relate to the optimisation process and the model training: optimization algorithm, learning rate, dropout rate, and regularisation.

The learning rate is a hyperparameter that determines how much the model's weights are updated after each iteration. L2 regularization (weight decay) is crucial because it prevents overfitting and therefore promotes a more generalizable model. Both learning rate and L2 regularization are target hyperparameters to be optimized.

There are two other key parameters that are not subject to optimization in this work. First, the type of optimization algorithm that is used to maximise the efficiency or reduce the error, depending on the parameters (weights and biases). The Adam (adaptive moment estimation) algorithm shows better performance compared to other optimization algorithms [54]. Finally, dropout rate is a regularization method used during training to reduce overfitting; for a wide range of networks and objectives, a value of 0.5 robustly approaches the optimal value [55].

Table 1 shows the four selected hyperparameters to be optimized and their respective search spaces.

Increasing dataset sizes and increasingly complex models are a major hurdle in hyperparameter optimization since they make closed box performance evaluation more expensive. Training a single hyperparameter configuration on large datasets can nowadays easily exceed several hours and take up to several days [56]. A common strategy to speed up manual tuning is therefore to probe an algorithm/hyperparameter configuration on a small subset of the data, by training it only for a few iterations, by running it on a subset of features, by only using one or a few of the cross-validation folds. Multi-fidelity methods cast such manual heuristics into formal algorithms, using so-called low-fidelity

approximations of the actual loss function to minimize. These approximations introduce a tradeoff between optimization performance and runtime, but in practice, the obtained speedups often outweigh the approximation error [57], [58]. For this target, in this work the optimization was implemented with 20% of the training data and only 100 epochs.

To take full advantage of BO, evaluations of the objective function must be performed. These evaluations allow exhaustively exploring the hyperparameter search space and finding the optimal combination that minimises the objective function. In this paper, the Root Mean Square Error (RMSE) is used as the criterion for selecting the optimal hyperparameter combination (see Figure 2), that is, during BO evaluations, the hyperparameter combination with the lowest RMSE is selected. Furthermore, to maintain consistency in the optimization process, in all experiments, the Mini Batch Size and time delay are set to one day (288 data) and 1 hour (12 data), respectively.

D. EVALUATION METRICS

To evaluate the performance of the proposed model, several error metrics are used [2]: the Root Mean Squared Error (RMSE), the Normalised Root Mean Squared Error (NRMSE), the Mean Absolute Error (MAE) and the Correlation Coefficient R^2, and can be expressed as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{9}$$

$$NRMSE = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}i)^2}}{y_{imax} - y_{imin}} \tag{10}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n} \mid y_i - \hat{y}_i \mid \tag{11}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{12}$$

where y_i and $\hat{y}_i$ are the actual and predicted values, respectively and n is the number of samples in the test data.

III. CASE OF STUDY

A. DATA AND BASE MODEL

The data used in this paper were collected in 2022 from the Desert Knowledge Australian Solar Centre (DKASC) project, with a temporal resolution of 5 minutes [59]. The project was established in 2008 at the Desert Knowledge Precinct of Alice Springs, a town in the Northern Territory located in the geographical centre of Australia. The DKASC has an installed capacity of 263 kW and consists of more than 40 installations from different manufacturers, making it the largest installation in the world for demonstrating multiple technologies of varying ages, brands, models, and configurations.

The downloaded data include meteorological variables such as ambient temperature ($°C$), relative humidity (%), global horizontal radiation (W/m^2), diffuse horizontal radiation (W/m^2), wind direction (°), and daily precipitation (mm). Additionally, total active power data (kW) are available.

The quality of the input data is a key factor in ensuring accurate and reliable forecasting performance. Numerous studies have highlighted the importance of using historical time series of PV output combined with meteorological variables and location-specific information to build predictive models. However, such datasets often contain noise, outliers, and missing values caused by weather variability, sensor errors, or power system interruptions, all of which can significantly affect model accuracy. Consequently, appropriate data pre-processing is essential to improve data quality, including outlier removal, missing data reconstruction, and normalization techniques [60].

The pre-processing and cleaning stage aimed to ensure data completeness and quality by removing entries with missing PV power values or associated meteorological measurements [61], [62]. The process involved several key steps:

1) Negative solar radiation values and missing PV power readings typically occurred during early morning and late evening hours, often due to sensor drift or inverter malfunctions. In such cases, both solar radiation and PV power values were set to zero.
2) Missing data during midday, such as solar radiation, temperature, or PV output, was likely caused by sensor failures or temporary disruptions in the inverter or power network. These records were excluded from the analysis.
3) To improve computational efficiency, preserve relationships among input variables, and facilitate rapid convergence in artificial neural networks (ANNs), the remaining dataset was normalized to the range [0, 1].

Several methods have been proposed in the literature for pre-processing and cleaning PV datasets, including wavelet transform (WT), empirical mode decomposition, singular spectrum analysis (SSA), normalization, Z-score and trend analysis [63]. In particular, Z-score is commonly applied to standardize input variables to have a mean of 0 and a standard deviation of 1, which enhances computational efficiency and preserves the relative correlation between variables [31], [64]. The technique used in this paper is Z-score standardization, which is performed by subtracting the mean from each value and dividing by the standard deviation, as shown in Equation 13.

$$X_{Standarization} = \frac{X - mean(X)}{standard\ deviation(X)} \tag{13}$$

Moreover, recent studies have applied additional techniques such as data filtering, weather classification, and even generative models (e.g., GANs) to handle missing data or augment rare events [61]. However, no single pre-processing technique can be considered universally optimal, as the most appropriate method strongly depends on the dataset characteristics and the specific forecasting task. Therefore,

the effectiveness of a pre-processing technique is typically evaluated based on several criteria, including computational time, resource usage, accuracy, data fidelity, performance consistency, adaptability, robustness, and compatibility with established forecasting models [61], [65].

In this study, missing data is handled using the K-Nearest Neighbour (KNN) algorithm. This algorithm replaces missing values of a specific variable with the mean of its neighbouring observations. Missing data values are estimated by calculating the distance between them and their nearest neighbours using the Euclidean distance formula [66], as shown in Equation 14.

$$D(x_m, x_o) = \sqrt{\sum_{i=1}^{n}(x_{mi} - x_{oi})^2} \tag{14}$$

where $D(x_m, x_o)$ represents the Euclidean distance between two data point in an n-dimensional space, x_{mi} is an existing data point (row i in the dataset), and x_{oi} is a data point whit a missing value whose distance to existing points is being computed.

To avoid redundant information and reduce training time, it is essential to select only the most relevant input features for the predictive model. However, since this approach does not capture non-linear dependencies or multicollinearity, two additional feature selection methods were employed to ensure a robust selection process: Mutual Information (MI), and Recursive Feature Elimination (RFE).

The Pearson Correlations Coefficient (PCC) was initially used to identify linear dependencies between each meteorological variable and the target output, Total Active Power. To account for potential non-linear relationships, the Mutual Information (MI) metric was applied. MI measures the amount of information obtained about one variable through another, capturing linear and non-linear dependencies. A high MI score indicates a strong dependency between variables, even if the relationship is non-linear [67]. Finally, Recursive Feature Elimination (RFE) was used as a wrapper method that iteratively removes the least important features based on the performance of a core learning algorithm [68]. RFE takes into account feature interactions and model-specific importance, making it suitable for identifying features that most contribute to the model's predictive accuracy. Table 2 shows the results of all three methods, providing a more comprehensive and reliable approach for selecting the optimal subset of input variables for photovoltaic power prediction. Table 2 shows that global horizontal radiation has a stronger correlation with photovoltaic power compared to diffuse horizontal radiation, followed by ambient temperature.

For feature selection, the PCC was used in this study to measure the strength of linear relationships between input variables and the target output. This approach has been widely adopted in previous works due to its simplicity, low computational cost, and ease of interpretation [69], [70]. Although this method does not capture nonlinear dependencies or multicollinearity, several studies have demonstrated

TABLE 2. Correlations among Total Active Power (kW) and meteorological variables of DKASC project.

Features	Pearson correlation coefficient	Mutual information value	Recursive feature elimination
- Global Horizontal Radiation (W/m^2)	0.9844	1.7048	0.5730
- Diffuse Horizontal Radiation (W/m^2)	0.5491	0.9167	0.3028
- Weather Temperature ($^\circ C$)	0.4492	0.2351	0.1182
- Wind Direction (degrees)	0.0629	0.0507	0.0135
- Weather Daily Rainfall (mm)	-0.0433	0.0180	0.0
- Weather Relative Humidity (%)	-0.4745	0.2480	0.0

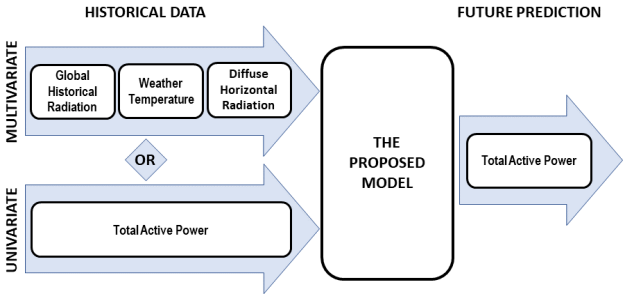


FIGURE 3. Schematic representation of univariate input and multivariate input configuration with a single output.

that key predictor variables in PV forecasting (such as solar irradiance, temperature, and time-based characteristics) are highly correlated with PV power output and have clear physical significance [71], [72].

Given the computational constraints, the three variables most correlated with photovoltaic energy were selected as input features in this work: ambient temperature ($^\circ C$), global horizontal radiation (W/m^2), and diffuse horizontal radiation (W/m^2). Likewise, the target variable to be predicted by the model is the Total Active Power (kW).

Another crucial aspect that has been considered in this paper is the input – output patterns. Two different approaches are investigated to understand these patterns to achieve accurate predictions. The first approach uses multivariate data, including the three most correlated variables mentioned above, as inputs and a single output variable, the total active power. In this multivariate approach, the goal is to analyse how weather inputs influence the prediction of photovoltaic power. The second approach is univariate, which means that a single input variable (the historical data of the total active power) is used to predict the same output variable. In this case, the main goal is to study how the historical data of the target variable affects its future prediction. This duality in approaches seeks to address photovoltaic power forecasting more comprehensively, considering both meteorological conditions and time trends and dependency relationships with historical

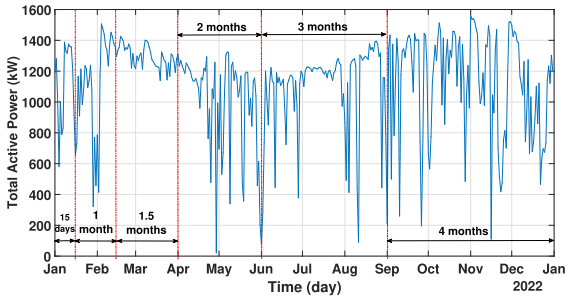


FIGURE 4. The active power series during the entire study period.

data, to obtain more accurate and complete predictions. Figure 3 shows how the input data is structured in the two approaches.

The data are converted to a daily resolution to visualize the behavior of the total active power during the study period, as shown in Figure 4, where power reaches its maximum in November (1516.36 kW) and its minimum at the end of Abril (21.21 kW). At the same time, it is observed that the total active power varies significantly over the study period. This variability may be due to sudden changes in weather conditions, which play an essential role in quantifying photovoltaic energy production.

Inputting a whole year's data (105120 points) at once into the LSTM model exponentially increases computational cost. The increase occurs because LSTM requires three-dimensional (3-D) input data [27]. Considering computational cost and aiming to study the entire time series and the effect of its length on performance, the data are divided into six DSS segments of different lengths, as shown in Figure 4.

- Fifteen days (4320 data points) correspond to the period from Jan. 1 to Jan. 15.
- One month (8928 data points) corresponds to the period from Jan. 16 to Feb. 15.
- One month and a half (12672 data points) correspond to the period from Feb. 16 to Mar. 31.
- Two months (17568 data points) correspond to the period from Apr. 1 to May 31
- Three months (26496 data points) correspond to the period from Jun. 1 to Aug. 31
- Four months (35136 data points) correspond to the period from Sep. 1 to Dec. 31

The data for each period are divided into two main groups: the first group is used for training (the first 80% of the data), and the second group is used for testing (the remaining 20%). The results are analyzed and discussed in the next subsection.

B. CASE 1

In this case study, the data were subjected to a cleaning and scaling process, as described in the subsection above. Then, two approaches were employed to select input features (univariate vs. multivariate). Subsequently, a Data Set Segmentation (DSS) was performed, varying the data length from 15 days to 4 months, as illustrated in Figure 4. Then, BO was applied to find the optimal combination

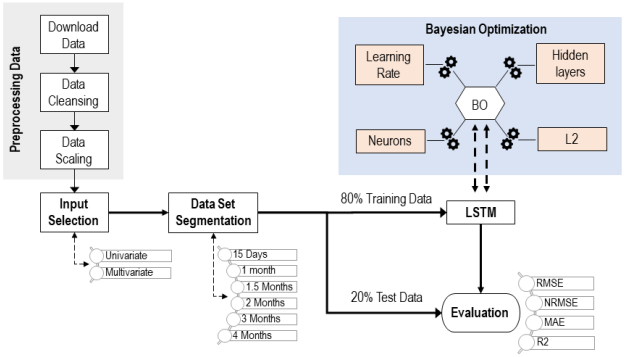


FIGURE 5. The overall architecture of case study 1.

TABLE 3. Optimal values of hyperparameters estimated by BO, depending on the DSS, for the univariate case.

Hyperparameter	15 Days	1 Month	1.5 Months	2 Months	3 Months	4 Months
Number of hidden layers	3	2	1	1	1	1
Number of neurons	75	105	99	186	199	115
Learning Rate	0.0033853	0.00338	0.00147	0.00218	0.0143	0.001003
L2 Regularization	$4.65{\cdot}10^{-10}$	$5.55{\cdot}10^{-9}$	$6.12{\cdot}10^{-8}$	$2.09{\cdot}10^{-10}$	$8.46{\cdot}10^{-7}$	$1.08{\cdot}10^{-10}$

TABLE 4. Evaluation of the accuracy of the LSTM model on both the training and test datasets for the univariate case, depending on the DSS.

Data set	15 Days		1 Month		1.5 Months		2 Months		3 Months		4 Months	
	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train
RMSE	12.464	9.773	6.268	5.761	7.365	5.936	9.344	7.29	7.423	5.805	13.304	10.544
NRMSE	0.249	0.201	0.111	0.106	0.138	0.109	0.249	0.162	0.145	0.107	0.339	0.217
MAE	5.149	3.736	2.912	1.928	2.77	2.083	3.367	2.525	2.387	1.737	5.932	3.962
R^2	0.981	0.988	0.996	0.996	0.994	0.996	0.987	0.993	0.994	0.996	0.969	0.987

of hyperparameters of the LSTM network, as detailed in subsection II-C, considering the input features and the segmented data. Finally, the LSTM network was trained on the training data set and evaluated using a test data set considering four error metrics described in subsection II-D. The overall architecture of the prediction model for case study 1 is illustrated in Figure 5.

For the univariate analysis, only the historical data of the total active power (kW) were considered to predict the next value (one-step) of the total active power. Table 3 shows the optimal values of the hyperparameters estimated by BO, depending on the DSS. The values of these hyperparameters were used to train the LSTM, depending on the DSS, to achieve better accuracy. Table 4 provides detailed performance metrics of the LSTM model on both the training and test datasets, depending on the DSS.

The multivariate analysis includes the three historical variables most correlated with the total active power for its future prediction (one-step): global horizontal radiation (W/m^2), diffuse horizontal radiation (W/m^2), and ambient temperature ($°C$), as input features. The target variable to be predicted by the model is the total active power (kW). Table 5 shows the optimal hyperparameter values estimated by BO, depending on the DSS. As in the univariate case, these hyperparameters values were used to train the LSTM network depending on the DSS. Table 6 provides a detailed performance analysis of the LSTM model on both the training and test datasets, depending on the DSS.

The impact of input length and type (univariate vs. multivariate) on prediction accuracy can be observed from

TABLE 5. Optimal values of hyperparameters estimated by BO, depending on the DSS, for the univariate case.

Hyperparameter	15 Days	1 Month	1.5 Months	2 Months	3 Months	4 Months
Number of hidden layers	3	3	2	3	3	3
Number of neurons	89	82	200	81	110	76
Learning Rate	0.001196	0.00724	0.00302	0.00218	0.003345	0.001614
L2 Regularization	0.0097	2.11E-8	1.67E-7	2.09E-10	1.29E-9	1.82E-10

TABLE 6. Evaluation of the accuracy of the LSTM model on both the training and test datasets for the multivariate case, depending on the DSS.

Data set	15 Days		1 Month		1.5 Months		2 Months		3 Months		4 Months	
	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train
RMSE	14.641	12.62	7.268	5.404	8.423	5.257	9.783	6.838	7.862	5.788	13.432	9.969
NRMSE	0.316	0.261	0.133	0.099	0.159	0.096	0.256	0.153	0.149	0.131	0.351	0.205
MAE	8.178	7.689	3.833	2.657	3.79	2.282	4.147	2.863	3.263	2.536	6.411	4.12
R^2	0.974	0.981	0.995	0.997	0.992	0.997	0.986	0.994	0.994	0.996	0.969	0.989

Tables 4 and 6. The results for each input dataset are analyzed below:

- When the input duration is 15 days, the accuracy of the LSTM model is poor for both univariate and multivariate data. This is due to the undertraining of the model.
- When the input size is one month, the results improve significantly, and the LSTM model achieves its highest accuracy compared to the other input durations, using both univariate and multivariate data.
- The results worsen for input sizes of 1.5, 2 and 4 months compared to the immediately preceding input size.
- For the 3-month input size, the RMSE, NRMSE, and MAE errors decrease again in both cases (univariate and multivariate).
- For all DSS sizes, both with univariate and multivariate data, the regression coefficient ranges between 0.969 and 0.996.

The results suggest that the accuracy of the LSTM model does not depend solely on increasing the data size. This could be due to the nature of the data itself and how it behaves over time. Photovoltaic power generation time series are highly variable, stochastic, and intermittent. Moreover, increasing the amount of data does not necessarily lead to better accuracy in the LSTM model. Increasing the data size beyond a certain point worsens the results, particularly with an input size of 4 months. This indicates that the LSTM network struggles to accurately capture both long- and short-term dependencies, particularly for photovoltaic data with a 5-minute resolution.

Regarding the model's predictive ability across different input lengths and types, in general, the closer the error values are between the training and test datasets, the better the model's predictive performance. As shown in Tables 4 and 6, for all cases (various DSS input lengths and input types, univariate or multivariate), the RMSE, NRMSE and MAE values in the training dataset closely match those in the test dataset and follow a similar trend, indicating consistent and stable model predictions.

C. CASE 2

In this case study, the FSST was incorporated as a feature extraction method for data input, as illustrated in Figure 6. The same input types (univariate vs. multivariate) and input lengths (DSS from 15 days to four months) as in Case 1 were used, along with the same hyperparameters. This was done to enable a proper comparison and assess whether using FSST improves the results obtained.

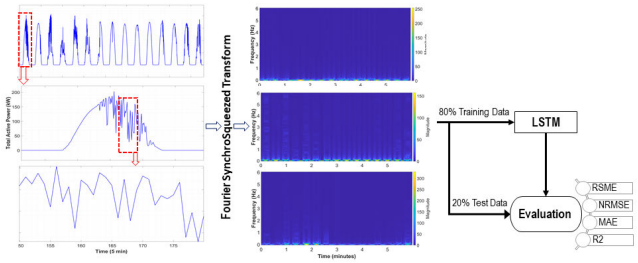


FIGURE 6. The overall architecture of case study 2.

TABLE 7. Evaluation of the accuracy of the proposed model on both the training and test datasets for the univariate case, depending on the DSS.

Data set	15 Days		1 Month		1.5 Months		2 Months		3 Months		4 Months	
	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train
RMSE	8.776	3.573	3.688	3.024	4.343	1.051	5.109	2.018	4.463	2.503	4.727	1.257
NRMSE	0.175	0.073	0.067	0.068	0.082	0.019	0.135	0.045	0.089	0.057	0.121	0.026
MAE	4.383	2.03	2.357	1.701	1.986	0.588	2.404	1.187	2.285	1.567	2.612	0.732
R^2	0.991	0.999	0.999	0.999	0.998	1	0.996	1	0.998	0.999	0.996	1

TABLE 8. Evaluation of the accuracy of the proposed model on both the training and test datasets for the multivariate case, depending on the DSS.

Data set	15 Days		1 Month		1.5 Months		2 Months		3 Months		4 Months	
	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train
RMSE	11.789	9.37	5.495	3.942	6.698	2.343	7.545	3.062	6.45	3.053	8.043	2.87
NRMSE	0.253	0.194	0.098	0.089	0.124	0.043	0.194	0.069	0.127	0.069	0.213	0.059
MAE	8.092	6.624	3.178	2.319	3.764	1.455	3.585	1.641	3.104	1.832	4.609	1.622
R^2	0.987	0.991	0.997	0.998	0.996	0.999	0.993	0.999	0.996	0.999	0.99	0.999

Table 7 presents the performance of the proposed model with univariate data on both training and the test datasets, depending on the DSS.

Table 8 presents the performance of the proposed model with multivariate data on both the training and test datasets, depending on the DSS.

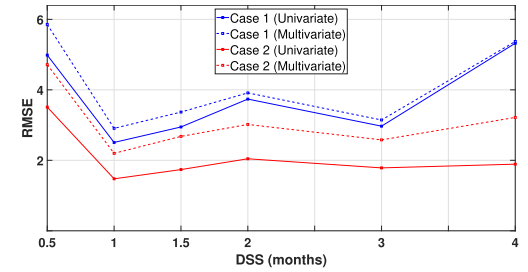
The comparison between Tables 7 and 8, in contrast to Tables 4 and 6, shows a significant improvement in prediction accuracy in Case Study 2 compared to Case Study 1, Figure 7. This improvement is consistent accross different durations and input types (univariate versus multivariate) and is especially significant for the 4-month input duration.

In summary, including the FSST in the data preparation phase for feeding the LSTM network proves beneficial for capturing the long- and short-term dependencies more accurately compared to Case Study 1.

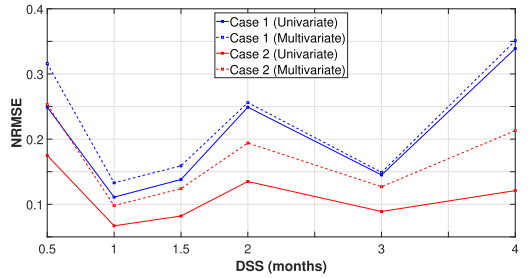
About the type of input (univariate versus multivariate) and its impact on prediction performance, Tables 7 and 8 show that RMSE, NRMSE, MAE and R^2 values are relatively close across all DSS. However, in the univariate case, the results exhibit more favorable error metrics.

D. CASE 3: FUTURE PREDICTION

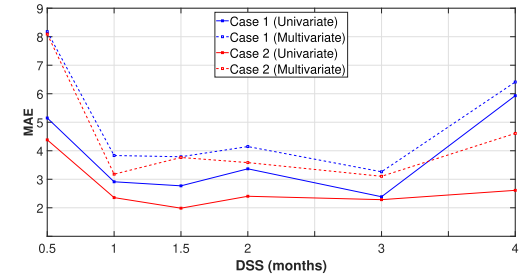
To evaluate the accuracy of the proposed BO-FSST-LSTM model at different prediction horizons, for each DSS (15 days, one month, one and a half months, two months, three months, and four months), prediction horizons of five minutes,



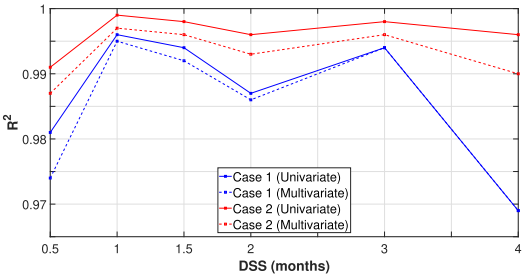
(a) RMSE.



(b) NRMSE.



(c) MAE.



(d) R2.

FIGURE 7. Error metrics vs. DSS (months) comparison for Case 1 and 2.

30 minutes, 1 hour, 6 hours, 1 day and 3 days were used. This process is presented graphically in Figure 8.

Table 9 shows that the BO-FSST-LSTM model delivers excellent prediction performance accross all prediction horizons and DSS, maintaning strong consistency even as the horizon increases.

From Table 9, the proposed BO-FSST-LSTM model shows great power to provide excellent prediction results for all prediction horizons and all DSS, especially consistent as the prediction horizon increases.

Figures 9(a) to 9(c) illustrate how RMSE values for training and test data vary across the different prediction

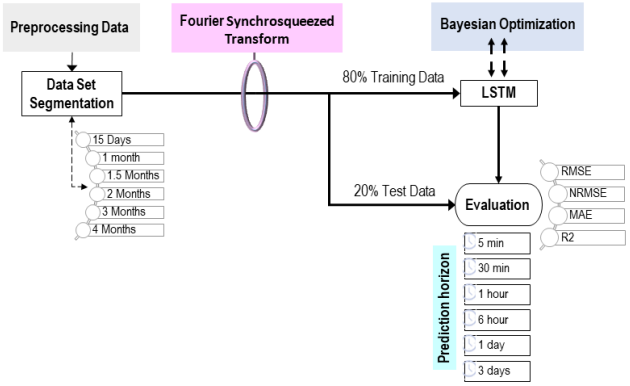


FIGURE 8. The overall architecture of case study 3.

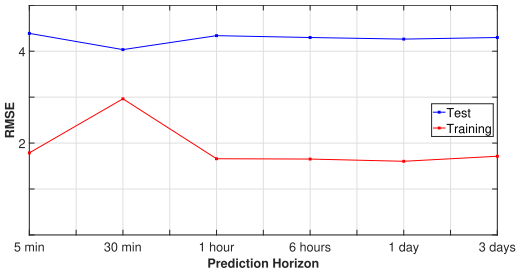
TABLE 9. Evaluation of the accuracy of the proposed model on the test and training datasets, depending on the DSS and varying the prediction horizon.

Period	Data set	Prediction horizon											
		5 min		30 min		1 hour		6 hour		1 day		3 days	
		Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train
15 Days	RMSE	8.776	3.573	8.069	5.927	8.677	3.318	8.596	3.304	8.527	3.207	8.596	3.426
	NRMSE	0.175	0.073	0.17	0.127	0.173	0.068	0.175	0.069	0.172	0.067	0.175	0.071
	MAE	4.383	2.03	4.66	3.842	4.479	2.034	4.29	1.906	4.138	1.832	4.63	2.236
	R^2	0.991	0.999	0.992	0.996	0.991	0.999	0.991	0.99	0.991	0.999	0.991	0.999
1 Month	RMSE	3.688	3.024	4.153	2.787	3.899	3.333	3.457	2.151	4.18	3.269	4.092	3.287
	NRMSE	0.067	0.068	0.073	0.063	0.071	0.075	0.06	0.048	0.074	0.074	0.074	0.075
	MAE	2.357	1.701	2.9	1.988	2.291	2.006	2.24	1.287	2.496	2.054	3.104	2.554
	R^2	0.999	0.999	0.999	0.999	0.999	0.999	0.999	1	0.999	0.999	0.999	0.999
1.5 Months	RMSE	4.343	1.051	3.964	1.367	4.837	1.007	4.464	1.079	4.223	0.793	3.98	2.368
	NRMSE	0.082	0.019	0.074	0.025	0.091	0.018	0.084	0.02	0.08	0.015	0.074	0.043
	MAE	1.986	0.588	2.033	0.893	2.271	0.635	1.907	0.562	1.882	0.475	2.586	1.918
	R^2	0.998	1	0.998	1	0.997	1	0.998	1	0.998	1	0.998	1
2 Months	RMSE	5.109	2.018	4.663	1.233	5.352	1.549	5.161	1.655	5.341	1.909	4.341	1.278
	NRMSE	0.135	0.045	0.124	0.027	0.143	0.034	0.138	0.037	0.142	0.042	0.116	0.028
	MAE	2.404	1.187	2.092	0.746	2.507	0.976	2.305	0.867	2.517	1.115	1.942	0.662
	R^2	0.996	1	0.997	1	0.996	1	0.996	1	0.996	1	0.997	1
3 Months	RMSE	4.463	2.503	4.757	2.716	5.295	2.924	4.564	1.271	4.624	1.209	4.772	2.776
	NRMSE	0.089	0.057	0.097	0.062	0.108	0.067	0.09	0.028	0.092	0.027	0.098	0.065
	MAE	2.285	1.567	2.926	1.998	3.253	2.267	1.836	0.705	1.936	0.723	2.976	2
	R^2	0.998	0.999	0.998	0.999	0.997	0.999	0.998	1	0.998	1	0.998	0.999
4 Months	RMSE	4.727	1.257	4.506	1.794	4.894	1.264	4.752	1.185	4.973	1.338	4.548	1.513
	NRMSE	0.121	0.026	0.115	0.037	0.126	0.026	0.123	0.024	0.128	0.028	0.116	0.031
	MAE	2.612	0.732	2.472	0.963	2.844	0.842	2.612	0.719	2.711	0.79	2.571	0.936
	R^2	0.996	1	0.997	1	0.996	1	0.996	1	0.996	1	0.996	1

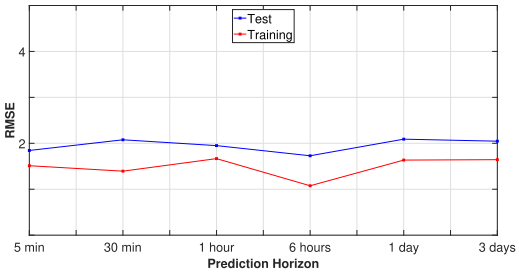
horizons (five minutes, 30 minutes, 1 hour, 6 hours, 1 day, and 3 days) for each DSS.

Regarding the 15-day DSS, Figure 9(a), RMSE, NRMSE, and MAE values for the training dataset remain consistently lower than those for the test dataset. Moreover, these error metrics follow the same trend as the prediction horizon increases. However, at the 30-minute horizon, the gap between training and test RMSE, NRMSE, and MAE values narrows. Therefore, the model exhibits greater stability at this horizon compared to others. For the one-month DSS, Figure 9(b), the RMSE, NRMSE, and MAE values follow a similar pattern to the previous case. The prediction accuracy remains almost constant as the prediction horizon increases. Furthermore, the difference in RMSE, NRMSE, and MAE values between training and test datasets are smaller compared to the 15-day DSS. The best performance for this DSS was observed at a 6-hour prediction horizon.

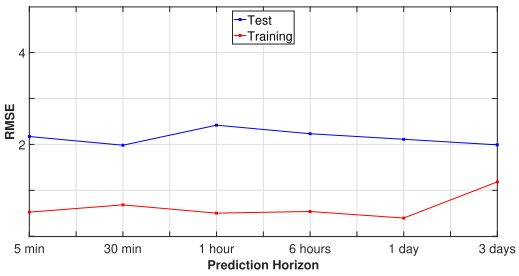
For the one and a half month DSS, Figure 9(c), RMSE, NRMSE, and MAE values for both training and test datasets show a pattern similar to that observed in Figure 9(a) (15-day DSS) as the prediction horizon varies. The RMSE, NRMSE, and MAE values in the test set, from a 1-hour time horizon, have a decreasing trend as the prediction horizon is increased from 1 hour to 3 days. In this case, the best results ocurred



(a) 15 days.

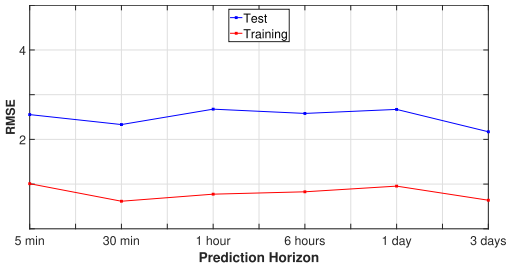


(b) 1 month.

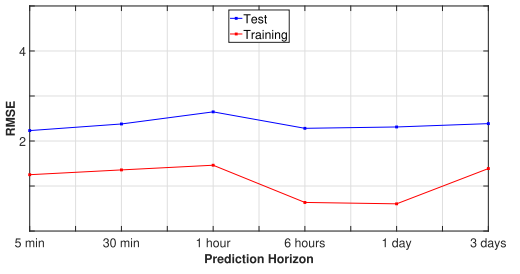


(c) 1 month and a half.

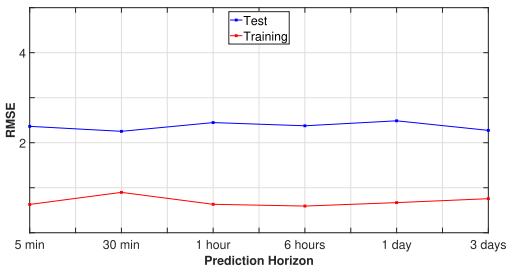
FIGURE 9. RMSE in training and test data varying the prediction horizon.



(a) 2 months.

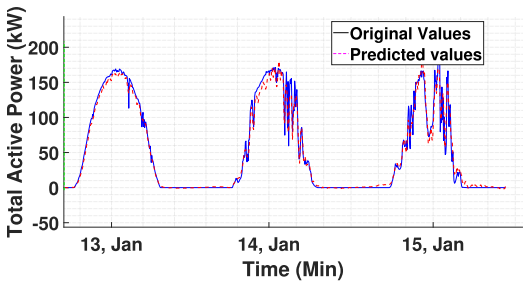


(b) 3 months.

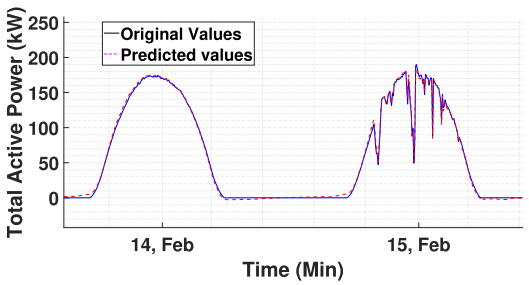


(c) 4 months.

FIGURE 10. RMSE in training and test data varying the prediction horizon.



(a) Jan. 13 (sunny), and 14-15 (sunny-cloudy).



(b) Feb. 14 (sunny), and 15 (cloudy).

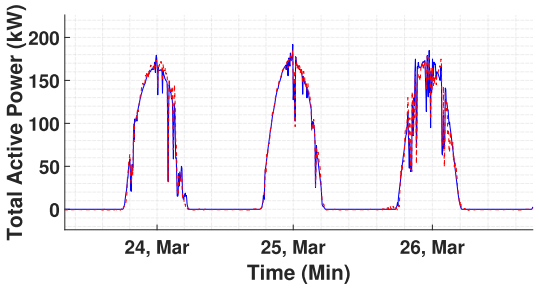
FIGURE 11. Original vs. predicted values.

at the 3-day prediction horizon, while the worst performance was seen at 1 hour.

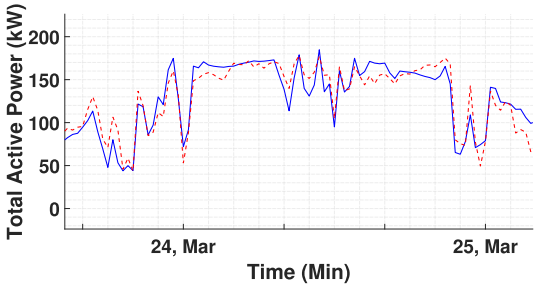
Similarly, for DSS of two months, three, and four months Figure 10(a) to Figure 10(c), RMSE, NRMSE, and MAE values for both training and test datasets follow a similar pattern to the previous cases. To summarize, the error metrics in Table 9 demonstrate the robustness of the proposed model across different scenarios, DSS lengths throughout the year Figure 4, and a wide range of prediction horizons.

Figure 11 compares original and predicted values for selected days with varying characteristics in the test dataset. These days exhibit different levels of stability in photovoltaic power production, including sunny, sunny-to-cloudy, cloudy, rainy, rainy-to-cloudy conditions. These figures show that the predicted values closely match the original trends, demonstrating the model's effectiveness across different DSS and prediction horizons.

For the one-month DSS, Figure 9(b), the RMSE, NRMSE and MAE values follow the same pattern as in the previous case. The prediction accuracy remains almost constant as the prediction horizon increases. Furthermore, the difference in RMSE, NRMSE and MAE values between the training and test datasets is smaller than in the 15-day DSS. For instance, at the 1-hour prediction horizon (best case), the RMSE, NRMSE and MAE differences are 0.566, 0.004 and 0.285,

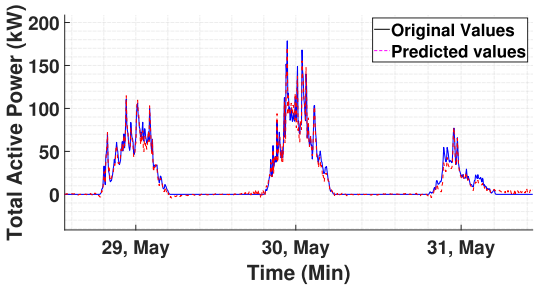


(a) Mar. 24-25 (sunny-cloudy) and 26 (cloudy).

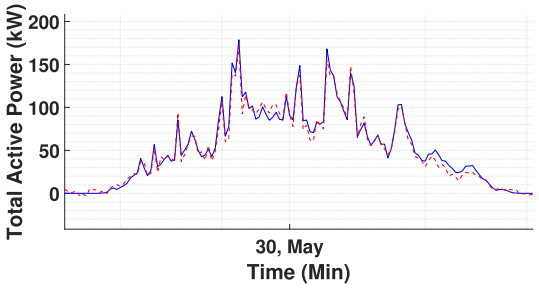


(b) Detail on Mar. 26 (cloudy).

FIGURE 12. Original vs. predicted values.



(a) May 29 (rainy), 30 (rainy-cloudy) and 31 (very rainy).

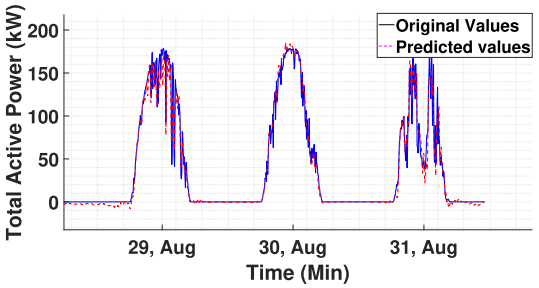


(b) Detail on May 30 (rainy-cloudy).

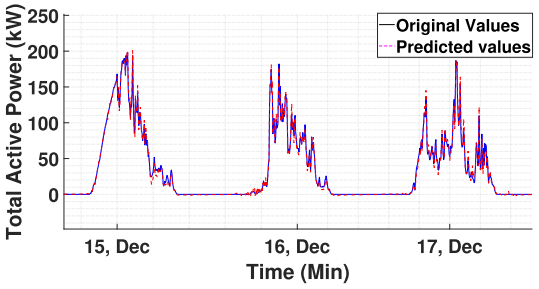
FIGURE 13. Original vs. predicted values.

respectively. In the worst case, at a 30-minute prediction horizon, the differences in RMSE, NRMSE, and MAE values are 1.366, 0.01, and 0.912, respectively. The R^2 value reamins constant at 0.999 in the test dataset. The best results were obtained at a prediction horizon of 6 hours in this data length.

For the month and a half case Figure 9(c), the RMSE, NRMSE and MAE values of the training and test datasets show a similar pattern to that shown in Figure 9(a) (the data length of 15 days) as the prediction horizon varies. The RMSE, NRMSE and MAE values in the test set, from



(a) Aug. 29 (partially sunny-cloudy), 30 (cloudy) and 31 (rainy).



(b) Dec. 15 (sunny-cloudy-rainy), 16 (rainy), and 17 (rainy).

FIGURE 14. Original vs. predicted values.

TABLE 10. Analysis of BO execution time (in seconds).

Data segmentation	Datapoints	Univariate	Multivariate
15 Days	864	78.4376	142.2385
1 Month	1786	153.8394	258.4824
1.5 Months	2534	266.7355	428.3773
2 Months	3514	363.5299	538.3478
3 Months	5300	579.8965	787.6884
4 Months	7028	828.9583	1347.991

TABLE 11. Analysis of training time (in seconds).

Data segmentation	Datapoints	Univariate	Multivariate
15 Days	4320	0:00:17	0:00:30
1 Month	8928	0:00:37	0:04:03
1.5 Month	12672	0:01:07	0:06:52
2 Months	17568	0:01:13	0:10:00
3 Months	26496	0:01:36	0:11:19
4 Months	35136	0:01:55	0:14:54

TABLE 12. Training time at different prediction horizons.

Data segmentation	Data points	5 Min	30 Min	1 Hour	6 Hours	1 Day	3 Days
15 Days	4320	0:00:17	0:01:42	0:02:50	0:12:15	0:50:27	2:25:05
1 Month	8928	0:00:37	0:02:50	0:04:29	0:23:18	2:29:03	5:00:29
1.5 Months	12672	0:01:07	0:03:58	0:06:26	0:35:08	2:48:25	8:00:29
2 Months	17568	0:01:13	0:06:48	0:11:25	0:41:34	3:13:30	9:30:27
3 Months	26496	0:01:36	0:10:53	0:15:57	1:09:45	4:58:03	14:13:21
4 Months	35136	0:01:55	0:12:19	0:18:47	1:23:03	6:20:24	20:54:17

1-hour time horizon, have a decreasing trend as the prediction horizon is increased from 1 hour to 3 days. In this case (one month and a half months), the best performance was achieved at a 3-day horizon ($R^2 = 0.998$), with RMSE, NRMSE and MAE values of 1.612, 0.031 and 0.668, respectively. While the worst results were obtained at a prediction horizon of 1 hour ($R^2 = 0.997$), yielding a difference in RMSE, NRMSE and MAE values of 3.83, 0.073 and 1.636, respectively.

TABLE 13. Comparison of related approaches for power prediction.

Author	Models	Data and prediction horizon	Performance
Sharadga et al. (2025) [24]	Three statistical models (ARMA, ARIMA and SARIMA) and six different types of NN models (BiLSTM, LSTM, fuzzy, c-mean clustering, LRNN, and MLP)	A grid-connected PV power plant in China with three prediction horizons (1 hour, 2 hours and 3 hours).	Using BiLSTM for a 1 hour ahead $\mathbf{RMSE} = 0.791$ $\mathbf{R^2} = 0.98$
Wang et al. (2020) [20]	Ensemble model LSTM-RNN with time correlation principle and three standard forecasting models BPNN and persistent model	Earth System Research Lab (ESRL) Global Monitoring Division website in Nevada, USA. A day ahead prediction.	Using LSTM-RNN for a 1 day ahead $\mathbf{RMSE} = 0.0568$ $\mathbf{MAE} = 0.0235$ $\mathbf{R^2} = 0.9776$
Mellit et al. (2021) [28]	LSTM, GRU, CNN1D, BiLSTM, Bi-GRU, CNN1D-LSTM, CNN1D-GRU	A data logger with different prediction horizons: 1 min, 5 min, 30 min, and 1 hour.	Using LSTM for a 1 min ahead $\mathbf{RMSE} = 0.16$ $\mathbf{MAE} = 0.05$ $\mathbf{R^2} = 0.992$
Paulov et al. (2024) [26]	LSTM optimized using a modified RSA and enhanced by a VMD and Empirical Mode	Two wind Power plants with hourly resoluction and three predictions horizons: 1 step, 2 steps, and 3 steps.	Using LSTM with modified RSA and VMD for a 1 step ahead $\mathbf{MSE} = 0.006701$ $\mathbf{R^2} = 0.9319$
Stojkovic et al. (2025) [33]	Light Weight LSTM with and without attention mechanisms optimized using a modified PSO.	Two grid-connected PV power plants in India and one PV power plant of the Institute Mihailo Pupin in Serbia with multi-step horizon and near future prediction.	Using LSTM optimized by modified PSO $\mathbf{RMSE} = 0.043065$ $\mathbf{MAE} = 0.018280$ $\mathbf{R^2} = 0.893008$ $\mathbf{MSE} = 0.001855$
Present work (2025)	A hybrid model LSTM networks, with hyperparameters optimized using BO, and combining with FSST.	Desert Knowledge Australian Solar Centre PV power plant, segmented into six different time windows: (15 days, 1, 1.5, 2, 3 and 4 months, and six diferent prediction horizons: 5 min, 30 min, 1 hour, 6 hours, 1 day and 3 days	Using FSST-LSTM-BO with 1 month data length and 6 hour prediction horizon $\mathbf{RMSE} = 3.457\ (3.688 - 8.776)$ $\mathbf{MAE} = 2.24\ (1.836 - 4.66)$ $\mathbf{R^2} = 0.999\ (0.991 - 0.999)$

Similarly, for DSS of two, three, and four months, Figure 10(a) to Figure 10(a), RMSE, NRMSE and MAE values in both training and test datasets follow a similar trend as in the 15-day, one month and one-and-a half month cases. In summary, the error metrics in Table 9 highlight the model's robustness across various DSS lenghts, Figure 4, seasonla conditions, and a wide range of prediction horizons.

Figure 11 to Figure 14 compare original and predicted values on selected test days with different prediction horizons and segmentation. These days present different types of stability and instability in photovoltaic power production, such as sunny days, sunny to cloudy days, cloudy days, rainy days, rainy to cloudy days, etc. These figures confirm that predicted trends closely follow the original values, demonstrating the model's effectiveness across various DSS lengths and for prediction horizons.

E. COMPUTATIONAL ANALYSIS

All calculations were performed in MATLAB on a personal laptop equipped with a 12th Gen Intel™ i7-1255U CPU (1.7 GHz), 16 GB of RAM, and integrated Intel™Xe graphics. Table 10 summarises the BO runtiem for both univariate and multivariate data across different DSS. Table 11 presents the training time for a 5-minute prediction horizon using univariate and multivariate data across various DSS configurations. Table 12 reports the training time using univariate data while varying the prediction horizon from 5 minutes to 3 days, under different DSS configurations.

As shown in Table 10, BO applied to multivariate data consistently requires more computational resources than in the univariate case across all DSS. Moreover, the same table shows that increasing the DSS size leads to longer BO runtimes for both univariate and multivariate cases. Notably, BO was executed on a reduced subset of the training dataset (20%), limited to 100 epochs, and optimized just four hyperparameters (see Table 11).

Table 11 shows that, across all DSS, training the network with multivariate data takes more time than with univariate data, consistent with the BO results. Additionally, training time increases with DSS size for both univariate and multivariate cases. In summary, the computational cost of both BO and network training is strongly influenced by the number of data points used. Furthermore, the number of neurons, learning rate, and number of hidden layers also affect computational efficiency.

Table 12 shows that computation time increases with the prediction horizon, from five minutes up to three days, across all DSSs. Likewise, for a fixed prediction horizon, larger DSS sizes also lead to longer computation times. Regardless of model accuracy, a viable forecasting system must produce predictions faster than the prediction horizon. This requirement was met in all cases, as the prediction time remained below the corresponding horizon across all DSSs.

Table 13 presents a selection of proven approaches for renewable energy prediction. For each study, the developed techniques, the type of data used, and the evaluation metrics are described. The table distinguishes between metrics

used to select the best-performing model in each study and those used for comparison across different models. Comparison of these metrics shows that the proposed model performs robustly across all prediction horizons (5 minutes, 30 minutes, 1 hour, 6 hours, 1 day, and 3 days), and maintains this performance across different DSS configurations. Consequently, the proposed hybrid FSST-BO-LSTM model constitutes a significant advancement in both the consistency and accuracy of photovoltaic power forecasting.

IV. CONCLUSION

This study demonstrates the effectiveness of a novel hybrid model that integrates the Fourier SynchroSqueezed Transform (FSST) and a Long Short-Term Memory (LSTM) neural network optimized via Bayesian Optimization (BO) for photovoltaic (PV) power forecasting. The model was evaluated using various data set segmentations (DSS), ranging from 15 days to four months, along with two types of input: (i) univariate, using only historical PV power values, and (ii) multivariate, incorporating global horizontal radiation (W/m^2), diffuse horizontal radiation (W/m^2), and ambient temperature ($°C$).

The FSST was employed as a feature extractor, enhancing the quality of inputs fed into the LSTM. Experimental results demonstrate that FSST significantly improves forecasting performance across all time horizons and DSS configurations considered.

To maximize model performance, key LSTM hyperparameters—such as the number of hidden layers, number of neurons, learning rate, and L2 regularization— were optimized using BO. The optimization process considered DSS, input type, and forecast horizon, enabling a robust and tailored model architecture. In addition to prediction accuracy, computational efficiency was also assessed. The model consistently met the requirement that the prediction time remain below the forecast horizon, a critical requirement for real-time applications.

The proposed FSST-BO-LSTM model has shown consistent performance across a wide range of forecast horizons (from 5 minutes to 3 days) and DSS lengths (from 15 days to 4 months). This adaptability allows it to be applied in diverse use cases, including real-time grid operation, energy storage management, market participation, and long-term planning. Its robustness across different DSS further enhances its practical applicability by enabling flexible training based on short-term or long-term data availability, without compromising accuracy. In summary, the proposed hybrid model offers a computationally efficient and accurate solution for PV power forecasting, contributing to the reliable and economical integration of solar energy into the electrical grid.

Nonetheless, this study presents certain limitations. Long-term forecast horizons were not considered, restricting direct application to strategic planning. Moreover, validation using data from other geographic locations or renewable sources such as wind power was not performed. Although BO was employed to optimize the LSTM, a broader exploration of the hyperparameter space was not conducted. Similarly, FSST parameters were not tuned, despite the transform's strong performance. Future work could address these limitations by incorporating spatial and climatological data, exploring additional time-frequency transforms, and enhancing model interpretability. Such efforts may lead to further improvements in forecast accuracy and operational transparency.

REFERENCES

[1] M. Talaat, T. Said, M. A. Essa, and A. Y. Hatata, "Integrated MFFNN-MVO approach for PV solar power forecasting considering thermal effects and environmental conditions," *Int. J. Electr. Power Energy Syst.*, vol. 135, Feb. 2022, Art. no. 107570, doi: 10.1016/j.ijepes.2021.107570.

[2] F. Rodríguez, I. Azcárate, J. Vadillo, and A. Galarza, "Forecasting intra-hour solar photovoltaic energy by assembling wavelet based time-frequency analysis with deep learning neural networks," *Int. J. Electr. Power Energy Syst.*, vol. 137, May 2022, Art. no. 107777, doi: 10.1016/j.ijepes.2021.107777.

[3] T. Ahmad, H. Zhang, and B. Yan, "A review on renewable energy and electricity requirement forecasting models for smart grid and buildings," *Sustain. Cities Soc.*, vol. 55, Apr. 2020, Art. no. 102052, doi: 10.1016/j.scs.2020.102052.

[4] C. Arbizu-Barrena, J. A. Ruiz-Arias, F. J. Rodríguez-Benítez, D. Pozo-Vázquez, and J. Tovar-Pescador, "Short-term solar radiation forecasting by advecting and diffusing MSG cloud index," *Sol. Energy*, vol. 155, pp. 1092–1103, Oct. 2017, doi: 10.1016/j.solener.2017.07.045.

[5] A. Mellit, A. M. Pavan, E. Ogliari, S. Leva, and V. Lughi, "Advanced methods for photovoltaic output power forecasting: A review," *Appl. Sci.*, vol. 10, no. 2, p. 487, Jan. 2020, doi: 10.3390/app10020487.

[6] S. Atique, S. Noureen, V. Roy, V. Subburaj, S. Bayne, and J. Macfie, "Forecasting of total daily solar energy generation using ARIMA: A case study," in *Proc. IEEE 9th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Las Vegas, NV, USA, Jan. 2019, pp. 114–119, doi: 10.1109/CCWC.2019.8666481.

[7] C. Scott, M. Ahsan, and A. Albarbar, "Machine learning for forecasting a photovoltaic (PV) generation system," *Energy*, vol. 278, Sep. 2023, Art. no. 127807, doi: 10.1016/j.energy.2023.127807.

[8] C. Feng, Y. Liu, and J. Zhang, "A taxonomical review on recent artificial intelligence applications to PV integration into power grids," *Int. J. Electr. Power Energy Syst.*, vol. 132, Nov. 2021, Art. no. 107176, doi: 10.1016/j.ijepes.2021.107176.

[9] M. Husein and I.-Y. Chung, "Day-ahead solar irradiance forecasting for microgrids using a long short-term memory recurrent neural network: A deep learning approach," *Energies*, vol. 12, no. 10, p. 1856, May 2019, doi: 10.3390/en12101856.

[10] H. Wang, Z. Lei, X. Zhang, B. Zhou, and J. Peng, "A review of deep learning for renewable energy forecasting," *Energy Convers. Manage.*, vol. 198, Oct. 2019, Art. no. 111799, doi: 10.1016/j.enconman.2019.111799.

[11] Z. Guo, Z. Zhuang, H. Tan, Z. Liu, P. Li, Z. Lin, W.-L. Shang, H. Zhang, and J. Yan, "Accurate and generalizable photovoltaic panel segmentation using deep learning for imbalanced datasets," *Renew. Energy*, vol. 219, Dec. 2023, Art. no. 119471, doi: 10.1016/j.renene.2023.119471.

[12] X. Luo and D. Zhang, "An adaptive deep learning framework for day-ahead forecasting of photovoltaic power generation," *Sustain. Energy Technol. Assessments*, vol. 52, Aug. 2022, Art. no. 102326, doi: 10.1016/j.seta.2022.102326.

[13] F. Rodríguez, M. Genn, L. Fontán, and A. Galarza, "Very short-term temperature forecaster using MLP and N-nearest stations for calculating key control parameters in solar photovoltaic generation," *Sustain. Energy Technol. Assessments*, vol. 45, Jun. 2021, Art. no. 101085, doi: 10.1016/j.seta.2021.101085.

[14] A. Gensler, J. Henze, B. Sick, and N. Raabe, "Deep learning for solar power forecasting—An approach using AutoEncoder and LSTM neural networks," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Budapest, Hungary, Oct. 2016, pp. 2858–2865, doi: 10.1109/SMC.2016.7844673.

[15] C. Feng and J. Zhang, "SolarNet: A sky image-based deep convolutional neural network for intra-hour solar forecasting," *Sol. Energy*, vol. 204, pp. 71–78, Apr. 2020, doi: 10.1016/j.solener.2020.03.083.

[16] R. Damaševič ius, L. Jovanovic, A. Petrovic, M. Zivkovic, N. Bacanin, D. Jovanovic, and M. Antonijevic, "Decomposition aided attention-based recurrent neural networks for multistep ahead time-series forecasting of renewable power generation," *PeerJ Comput. Sci.*, vol. 10, p. e1795, Jan. 2024, doi: 10.7717/peerj-cs.1795.

[17] P. Soni, D. Mondal, and P. Mishra, "Fourier synchro squeezed-LSTM-CNN for categorization of power quality disturbances," in *Proc. IEEE 3rd Int. Conf. Smart Technol. Power, Energy Control (STPEC)*, Bhubaneswar, India, Dec. 2023, pp. 1–6, doi: 10.1109/stpec59253.2023.10431307.

[18] A. Kahraman, G. Yang, and P. Hou, "Wind power forecasting using LSTM incorporating Fourier transformation based denoising technique," in *Proc. 20th Int. Workshop Large-Scale Integr. Wind Power Power Syst. Well Transmiss. Netw. Offshore Wind Power Plants (WIW)*, Berlin, Germany, Sep. 2021, pp. 94–98, doi: 10.1049/icp.2021.2604.

[19] L. Wang, Y. Liu, T. Li, X. Xie, and C. Chang, "Short-term PV power prediction based on optimized VMD and LSTM," *IEEE Access*, vol. 8, pp. 165849–165862, 2020, doi: 10.1109/ACCESS.2020.3022246.

[20] J. Ospina, A. Newaz, and M. O. Faruque, "Forecasting of PV plant output using hybrid wavelet-based LSTM-DNN structure model," *IET Renew. Power Gener.*, vol. 13, no. 7, pp. 1087–1095, May 2019, doi: 10.1049/iet-rpg.2018.5779.

[21] J. Li, C. Zhang, and B. Sun, "Two-stage hybrid deep learning with strong adaptability for detailed day-ahead photovoltaic power forecasting," *IEEE Trans. Sustain. Energy*, vol. 14, no. 1, pp. 193–205, Jan. 2023, doi: 10.1109/TSTE.2022.3206240.

[22] F. Wang, Z. Xuan, Z. Zhen, K. Li, T. Wang, and M. Shi, "A day-ahead PV power forecasting method based on LSTM-RNN model and time correlation modification under partial daily pattern prediction framework," *Energy Convers. Manage.*, vol. 212, May 2020, Art. no. 112766, doi: 10.1016/j.enconman.2020.112766.

[23] H. Sharadga, S. Hajimirza, and R. S. Balog, "Time series forecasting of solar power generation for large-scale photovoltaic plants," *Renew. Energy*, vol. 150, pp. 797–807, May 2020, doi: 10.1016/j.renene.2019.12.131.

[24] M. A. F. B. Lima, P. C. M. Carvalho, L. M. Fernández-Ramírez, and A. P. S. Braga, "Improving solar forecasting using deep learning and portfolio theory integration," *Energy*, vol. 195, p. 45, Mar. 2020, Art. no. 117016, doi: 10.1016/j.energy.2020.117016.

[25] M. Pavlov-Kagadejev, L. Jovanovic, N. Bacanin, M. Deveci, M. Zivkovic, M. Tuba, I. Strumberger, and W. Pedrycz, "Optimizing long-short-term memory models via metaheuristics for decomposition aided wind energy generation forecasting," *Artif. Intell. Rev.*, vol. 57, no. 3, Feb. 2024, doi: 10.1007/s10462-023-10678-y.

[26] M. Abdel-Nasser and K. Mahmoud, "Accurate photovoltaic power forecasting models using deep LSTM-RNN," *Neural Comput. Appl.*, vol. 31, no. 7, pp. 2727–2740, Jul. 2019, doi: 10.1007/s00521-017-3225-z.

[27] A. Mellit, A. M. Pavan, and V. Lughi, "Deep learning neural networks for short-term photovoltaic power forecasting," *Renew. Energy*, vol. 172, pp. 276–288, Jul. 2021, doi: 10.1016/j.renene.2021.02.166.

[28] K. Wang, X. Qi, and H. Liu, "A comparison of day-ahead photovoltaic power forecasting models based on deep learning neural network," *Appl. Energy*, vol. 251, Oct. 2019, Art. no. 113315, doi: 10.1016/j.apenergy.2019.113315.

[29] X. Luo, D. Zhang, and X. Zhu, "Combining transfer learning and constrained long short-term memory for power generation forecasting of newly-constructed photovoltaic plants," *Renew. Energy*, vol. 185, pp. 1062–1077, Feb. 2022, doi: 10.1016/j.renene.2021.12.104.

[30] M. Massaoudi, I. Chihi, L. Sidhom, M. Trabelsi, S. S. Refaat, H. Abu-Rub, and F. S. Oueslati, "An effective hybrid NARX-LSTM model for point and interval PV power forecasting," *IEEE Access*, vol. 9, pp. 36571–36588, 2021, doi: 10.1109/ACCESS.2021.3062776.

[31] R. Ahmed, V. Sreeram, R. Togneri, A. Datta, and M. D. Arif, "Computationally expedient photovoltaic power forecasting: A LSTM ensemble method augmented with adaptive weighting and data segmentation technique," *Energy Convers. Manage.*, vol. 258, Apr. 2022, Art. no. 115563, doi: 10.1016/j.enconman.2022.115563.

[32] A. Stojkovic, B. Nikolic, M. Zivkovic, and N. Bacanin, "Photovoltaic farm production forecasting: Modified metaheuristic optimized long short-term memory-based networks approach," *IEEE Access*, vol. 13, pp. 25198–25222, 2025, doi: 10.1109/ACCESS.2025.3537407.

[33] G. Li, L. Yang, C.-G. Lee, X. Wang, and M. Rong, "A Bayesian deep learning RUL framework integrating epistemic and aleatoric uncertainties," *IEEE Trans. Ind. Electron.*, vol. 68, no. 9, pp. 8829–8841, Sep. 2021, doi: 10.1109/TIE.2020.3009593.

[34] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[35] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 6, no. 2, pp. 107–116, Apr. 1998, doi: 10.1142/s0218488598000094.

[36] C. Li and M. Liang, "A generalized synchrosqueezing transform for enhancing signal time–frequency representation," *Signal Process.*, vol. 92, no. 9, pp. 2264–2274, Sep. 2012, doi: 10.1016/j.sigpro.2012.02.019.

[37] A. Akan and O. Karabiber Cura, "Time–frequency signal processing: Today and future," *Digit. Signal Process.*, vol. 119, Dec. 2021, Art. no. 103216, doi: 10.1016/J.dsp.2021.103216.

[38] G. A. Campbell and R. M. Foster, "Fourier integrals for practical applications," in *Bell Telephone System Technical Publications, Mathematical Physics, Monograph B-584*. Princeton, NJ, USA, 1931, p. 177.

[39] J. Morlet, G. Arens, E. Fourgeau, and D. Glard, "Wave propagation and sampling theory—Part I: Complex signal and scattering in multilayered media," *Geophysis*, vol. 47, no. 2, pp. 203–221, Feb. 1982, doi: 10.1190/1.1441328.

[40] J. Ville, "Thorie et applications de la notion de signal analytique," *Cables et Transmissions*, vol. 2, no. 1, pp. 61–74, 1948. [Online]. Available: https://archive.org/details/VilleSigAnalytiqueCablesEtTrans1948Fr/mode/2up

[41] L. Stankovic, "A method for time-frequency analysis," *IEEE Trans. Signal Process.*, vol. 42, no. 1, pp. 225–229, Jun. 1994, doi: 10.1109/78.258146.

[42] H.-I. Choi and W. J. Williams, "Improved time-frequency representation of multicomponent signals using exponential kernels," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 6, pp. 862–871, Jun. 1989, doi: 10.1109/ASSP.1989.28057.

[43] S. Meignen, D.-H. Pham, and S. McLaughlin, "On demodulation, ridge detection, and synchrosqueezing for multicomponent signals," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 2093–2103, Apr. 2017, doi: 10.1109/TSP.2017.2656838.

[44] D. Iatsenko, P. V. E. McClintock, and A. Stefanovska, "Linear and synchrosqueezed time–frequency representations revisited: Overview, standards of use, resolution, reconstruction, concentration, and algorithms," *Digit. Signal Process.*, vol. 42, pp. 1–26, Jul. 2015, doi: 10.1016/j.dsp.2015.03.004.

[45] L. Li, H. Cai, H. Han, Q. Jiang, and H. Ji, "Adaptive short-time Fourier transform and synchrosqueezing transform for non-stationary signal separation," *Signal Process.*, vol. 166, Jan. 2020, Art. no. 107231, doi: 10.1016/j.sigpro.2019.07.024.

[46] R. G. Baraniuk and D. L. Jones, "Signal-dependent time-frequency analysis using a radially Gaussian kernel," *Signal Process.*, vol. 32, no. 3, pp. 263–284, Jun. 1993, doi: 10.1016/0165-1684(93)90001-q.

[47] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, Nov. 2020, doi: 10.1016/j.neucom.2020.07.061.

[48] W. Yue, Q. Liu, Y. Ruan, F. Qian, and H. Meng, "A prediction approach with mode decomposition-recombination technique for short-term load forecasting," *Sustain. Cities Soc.*, vol. 85, Oct. 2022, Art. no. 104034, doi: 10.1016/j.scs.2022.104034.

[49] M. K. Suddle and M. Bashir, "Metaheuristics based long short term memory optimization for sentiment analysis," *Appl. Soft Comput.*, vol. 131, Dec. 2022, Art. no. 109794, doi: 10.1016/j.asoc.2022.109794.

[50] N. Gorgolis, I. Hatzilygeroudis, Z. Istenes, and L. G. Gyenne, "Hyperparameter optimization of LSTM network models through genetic algorithm," in *Proc. 10th Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, Patras, Greece, Jul. 2019, pp. 1–4, doi: 10.1109/IISA.2019.8900675.

[51] Y. Mao, A. Pranolo, A. P. Wibawa, A. B. P. Utama, F. A. Dwiyanto, and S. Saifullah, "Selection of precise long short term memory (LSTM) hyperparameters based on particle swarm optimization," in *Proc. Int. Conf. Appl. Artif. Intell. Comput. (ICAAIC)*, Salem, India, May 2022, pp. 1114–1121, doi: 10.1109/ICAAIC53929.2022.9792708.

[52] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas, "Taking the human out of the loop: A review of Bayesian optimization," *Proc. IEEE*, vol. 104, no. 1, pp. 148–175, Jan. 2016, doi: 10.1109/JPROC.2015.2494218.

[53] M. G. Genton, "Classes of kernels for machine learning: A statistics perspective," *J. Mach. Learn. Res.*, vol. 2, no. 2, pp. 299–312, Mar. 2002. [Online]. Available: https://www.jmlr.org/papers/volume2/genton01a/genton01a.pdf

[54] D. P. Kingma, P. K. Diederik, and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent.*, San Diego, CA, USA, 2015.

[55] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, Jan. 2014. [Online]. Available: https://dl.acm.org/doi/abs/10.5555/2627435.2670313

[56] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 84–90, doi: 10.1145/3065386.

[57] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 281–305, Mar. 2012.

[58] F. Hutter, L. Kotthoff, and J. Vanschoren, *Automated Machine Learning Methods, Systems, Challenges. Hyperparameter Optimization*. Cham, Switzerland: SpringerLink, 2019, ch. 1, pp. 3–33, doi: 10.1007/978-3-030-05318-5.

[59] (2008). *DKA Solar Centre*. Accessed: Sep. 12, 2023. [Online]. Available: https://dkasolarcentre.com.au10.1109/ACCESS.2020.3022246

[60] Z. Yang and J. Wang, "A hybrid forecasting approach applied in wind speed forecasting based on a data processing strategy and an optimized artificial intelligence algorithm," *Energy*, vol. 160, pp. 87–100, Oct. 2018, doi: 10.1016/j.energy.2018.07.005.

[61] F. Wang, Z. Zhang, C. Liu, Y. Yu, S. Pang, N. Duić, M. Shafie-Khah, and J. P. S. Catalão, "Generative adversarial networks and convolutional neural networks based weather classification model for day ahead short-term photovoltaic power forecasting," *Energy Convers. Manage.*, vol. 181, pp. 443–462, Feb. 2019, doi: 10.1016/j.enconman.2018.11.074.

[62] M. K. Behera, I. Majumder, and N. Nayak, "Solar photovoltaic power forecasting using optimized modified extreme learning machine technique," *Eng. Sci. Technol., Int. J.*, vol. 21, no. 3, pp. 428–438, Jun. 2018, doi: 10.1016/j.jestch.2018.04.013.

[63] U. K. Das, K. S. Tey, M. Seyedmahmoudian, S. Mekhilef, M. Y. I. Idris, W. Van Deventer, B. Horan, and A. Stojcevski, "Forecasting of photovoltaic power generation and model optimization: A review," *Renew. Sustain. Energy Rev.*, vol. 81, pp. 912–928, Jan. 2018, doi: 10.1016/j.rser.2017.08.017.

[64] C. Yang, A. A. Thatte, and L. Xie, "Multitime-scale data-driven spatio-temporal forecast of photovoltaic generation," *IEEE Trans. Sustain. Energy*, vol. 6, no. 1, pp. 104–112, Jan. 2015, doi: 10.1109/TSTE.2014.2359974.

[65] F. Wang, K. Li, L. Zhou, H. Ren, J. Contreras, M. Shafie-Khah, and J. P. S. Catalão, "Daily pattern prediction based classification modeling approach for day-ahead electricity price forecasting," *Int. J. Electr. Power Energy Syst.*, vol. 105, pp. 529–540, Feb. 2019, doi: 10.1016/j.ijepes.2018.08.039.

[66] Y. Chu, B. Urquhart, S. M. I. Gohari, H. T. C. Pedro, J. Kleissl, and C. F. M. Coimbra, "Short-term reforecasting of power output from a 48 MWe solar PV plant," *Sol. Energy*, vol. 112, pp. 68–77, Feb. 2015, doi: 10.1016/j.solener.2014.11.017.

[67] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 69, no. 6, Jun. 2004, Art. no. 066138, doi: 10.1103/physreve.69.066138.

[68] W. Liu and J. Wang, "Recursive elimination–election algorithms for wrapper feature selection," *Appl. Soft Comput.*, vol. 113, Dec. 2021, Art. no. 107956, doi: 10.1016/j.asoc.2021.107956.

[69] G. G. Kim, J. H. Choi, S. Y. Park, B. G. Bhang, W. J. Nam, H. L. Cha, N. Park, and H.-K. Ahn, "Prediction model for PV performance with correlation analysis of environmental variables," *IEEE J. Photovolt.*, vol. 9, no. 3, pp. 832–841, May 2019, doi: 10.1109/JPHOTOV.2019.2898521.

[70] M. A. Khan, M. A. Khan, H. Ali, B. Ashraf, S. Khan, D.-E.-Z. Baig, A. Wadood, and T. Khurshaid, "Output power prediction of a photovoltaic module through artificial neural network," *IEEE Access*, vol. 10, pp. 116160–116166, 2022, doi: 10.1109/ACCESS.2022.3216384.

[71] H. Zhu, B. Zhang, W. Song, J. Dai, X. Lan, and X. Chang, "Power-weighted prediction of photovoltaic power generation in the context of structural equation modeling," *Sustainability*, vol. 15, no. 14, p. 10808, Jul. 2023, doi: 10.3390/su151410808.

[72] S. Al-Dahidi, B. Rinchi, M. Alrbai, and A. Alahmer, "Enhancing daily energy prediction in solar photovoltaic systems: Weighted k-nearest neighbors with Pearson correlation integration," in *Proc. IEEE 12th Int. Conf. Intell. Syst. (IS)*, Aug. 2024, pp. 1–6, doi: 10.1109/is61756.2024.10705205.

SAMER RAJAH received the B.Sc. degree in electrical engineering from the University of Aleppo, in 2016, and the master's degree in intelligent systems in energy and transport from the University of Málaga, in 2022. He is currently pursuing the joint Ph.D. degree BFH University and Delft University. He was a Research Fellow with the University of Málaga, from 2023 and 2024, respectively.

His main research interest include solar energy prediction, deep learning models, closed-box optimization, and signal analysis.

FRANCISCO J. MUÑOZ received the B.E. degree in electrical engineering, the B.E. degree in automation and industrial electronics, the M.E. degree in software engineering and artificial intelligence, and the Ph.D. degree in mechatronics engineering program from the University of Málaga. He is currently a Senior Lecturer in circuits, electromagnetism, and electrical efficiency with the University of Málaga, where he has held the academic positions of the Secretary of the Department of Electrical Engineering and Electronic Technology, from 1987 to 1993, the Deputy Director of the Faculty of Industrial Engineering, from 1996 to 2000, and the Director of this Faculty, from 2000 to 2010. He obtained a research grant from the University Institute of Research and Technological Control.

He includes in his research topics, transmission lines and artificial intelligence techniques applied to power systems, electromagnetic compatibility, electric vehicles and electricity market, where he has participated as a Researcher in several innovation, and knowledge transfer projects, contributing extensively with papers in internationally renowned journals, conferences, distributed books and patents. He has been the recipient of the Technological Innovation Prize. He is also a registered professional engineer, and has been distinguished as an honorary member.

ALEJANDRO RODRÍGUEZ received the B.E. degree in electrical engineering, the B.E. degree in industrial management engineering, the M.E. degree in software engineering and artificial intelligence, and the Ph.D. degree in production systems engineering from the University of Málaga. Before completing his undergraduate studies, he was hired as a Research Staff Member within a university-company agreement. He joined as a Graduate Teaching Assistant, in 1995, and as a Lecturer, in 2000. Since 2011, he has been a Senior Lecturer in circuits and analysis of electrical networks at the University of Málaga. He has extensive experience in university management, having been the Dean of the Faculty of Industrial Engineering, University of Málaga, from 2010 to 2024.

He has developed his research on power quality, electromagnetic transients and the application of time-frequency transforms, and artificial intelligence techniques in electrical systems. In these areas, he is the author of different bibliographic productions, which including research and development projects, book contributions, and papers in international journals and conferences. He has collaborated with power transmission and distribution companies, and has several patents. He is also the co-founder of a technology-based company.