Renewable Energy 257 (2026) 124738

Contents lists available at ScienceDirect

Renewable Energy

journal homepage: www.elsevier.com/locate/renene

Multi-step short-term solar energy forecasting using Fourier-enhanced BiLSTM and neural additive models☆

Laio Oriel Seman [a,*], Stefano Frizzo Stefenon [b,c], Kin-Choong Yow [c], Leandro dos Santos Coelho [d,e], Viviana Cocco Mariani [d,f]

[a] Department of Automation and Systems Engineering, Federal University of Santa Catarina (UFSC), Florianópolis, SC, Brazil
[b] Lisbon School of Engineering (ISEL), Polytechnic University of Lisbon (IPL), Lisbon, Portugal
[c] Faculty of Engineering and Applied Sciences, University of Regina (UofR), Regina, SK, Canada
[d] Department of Electrical Engineering, Federal University of Parana (UFPR), Curitiba, PR, Brazil
[e] Graduate Program in Electrical Engineering, Federal University of Parana (UFPR), Curitiba, PR, Brazil
[f] Graduate Program in Mechanical Engineering, Federal University of Parana (UFPR), Curitiba, PR, Brazil

ARTICLE INFO

Keywords:
Photovoltaic energy forecasting
Solar energy
Meteorological data
Multi-step forecasting
Neural additive model
Machine learning

ABSTRACT

Accurate short and medium-term forecasting is important for mitigating uncertainty and enabling efficient energy grid management. While traditional machine learning and deep learning models offer improved accuracy, they often lack interpretability. To address these limitations, this study proposes a hybrid forecasting framework, called FNO-BiLSTM-NAM, that combines a Fourier Neural Operator (FNO) to extract spectral–temporal features, a Bidirectional Long Short-Term Memory (BiLSTM) network to model sequential dependencies, and a Neural Additive Model (NAM) to quantify feature-wise contributions. The model incorporates multi-scenario forecasting to support energy operators under different uncertainty levels. Experiments conducted on a dataset from a 5 MW PhotoVoltaic (PV) plant demonstrate the superiority of the model. For a 6-hour forecast horizon, the proposed FNO-BiLSTM-NAM model achieved a mean absolute error of 0.0712 and mean squared error of 0.0092, outperforming benchmark models across short- to medium-term horizons. Furthermore, the spectral analysis of the FNO revealed low-pass filtering behavior, highlighting the ability of the model to suppress high-frequency noise. Comparative experiments with five machine and deep learning baseline models confirm the robustness and generalization capacity of the framework. These results underscore the potential of the proposed model for enhancing PV energy forecasting accuracy while maintaining transparency across dynamic operating conditions.

1. Introduction

PhotoVoltaic (PV) and wind energy are the leading sources of renewable electricity generation. However, their inherent variability and intermittency pose significant challenges to power system stability, making accurate forecasting essential for efficient grid integration [1]. Forecasting methods for wind and PV energy can be broadly classified into four categories: physical models, statistical approaches, Artificial Intelligence (AI)-based approaches including Machine Learning (ML) and Deep Learning (DL), and hybrid models [2].

Physical models rely on deterministic frameworks, using Numerical Weather Prediction (NWP) data and geospatial parameters to simulate wind speed or solar irradiance before converting these into energy forecasts. However, due to the limited temporal resolution of NWP updates, these methods are best suited for medium- to long-term forecasting. In contrast, statistical and AI-based techniques employ data-driven approaches, learning nonlinear relationships between historical power output, operational data, and meteorological variables to generate predictions [3]. These methods excel in very short- to short-term

☆ This study was partially supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), funding reference number DDG-2024-00035. Cette recherche a été financée par le Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG), numéro de référence DDG-2020-00034. The authors, Mariani and Coelho, thank the National Council of Scientific and Technological Development of Brazil - CNPq (Grants number: 314389/2023-7-PQ, 313169/2023-3-PQ, 407453/2023-7-Universal, and 442176/2023-6-Peci). Mariani would especially like to thank the company PLIN Energia, located in Loanda, Parana state, Brazil, for providing real data for this study.

* Corresponding author at: Department of Automation and Systems Engineering, Federal University of Santa Catarina (UFSC), Florianópolis, SC, Brazil.
E-mail addresses: laioseman@gmail.com (L.O. Seman), stefano.stefenon@isel.pt (S.F. Stefenon), Kin-Choong.Yow@uregina.ca (K.-C. Yow), leandro.coelho@ufpr.br (L.d.S. Coelho), viviana.mariani@ufpr.br (V.C. Mariani).

https://doi.org/10.1016/j.renene.2025.124738
Received 30 July 2025; Received in revised form 21 September 2025; Accepted 8 November 2025
Available online 10 November 2025
0960-1481/© 2025 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC license (http://creativecommons.org/licenses/by-nc/4.0/).

forecasting. Hybrid models combine multiple methodologies, leveraging their complementary strengths to improve both robustness and accuracy [4].

Among various hybrid methods, Kushwaha and Pindoriya [5] proposed a model that combines Seasonal AutoRegressive Integrated Moving Average (SARIMA) with a Random Vector Functional Link (RVFL) neural network approach, enhanced by maximum overlap discrete wavelet transform for time-adaptive decomposition of time series of solar PV power using data from rooftop solar plants, showing improvements in forecast accuracy and adaptability compared to individual models. Tavares et al. [6] evaluated the accuracy of multilayer feed-forward Artificial Neural Networks (ANNs) and Deep Neural Networks (DNNs) to forecast PV generation of a residential building. The prediction was made 7 days ahead, and the ANN gives the most acceptable forecast errors.

An ensemble learning model was proposed by Khan et al. [7] utilizing two DL algorithms, ANN and Long Short-Term Memory (LSTM) as base models integrated with an eXtreme Gradient Boosting (XGBoost) algorithm for solar energy forecast. Ma et al. [8] proposed a DL model for PV power prediction based on Gated Recurrent Unit (GRU) network, improved sine cosine optimization algorithm, and full-set empirical mode decomposition which was applied to PV power data with good prediction results. Five different ML models were applied by Pombo et al. [9] to the short-term PV forecast. Solar irradiance, another important variable, was predicted 10 min ahead using LSTM and Convolutional Neural Network (CNN) [10]. Due to its immense potential, a wavelet-based time–frequency analysis of data used with a DL model to predict solar irradiance in the next 10 min to calculate solar PV generation was proposed by Rodríguez et al. [11].

Wang et al. [12] presented a study considering the accuracy rate, utility, and stability of the forecast, reaching the Pareto optimal solution. Four Belgian databases from one year were used. A model based on an Improved Stacking ensemble LSTM-Informer algorithm (ISt-LSTM-Informer) was applied by Cao et al. [13] to accurately short- and medium-term PV power forecasts using a dataset from a PV system located in Australia, with superior results. Jacques Molu et al. [14] proposed an approach for short-term solar power forecasting, integrating Bayesian optimized Bidirectional LSTM (BiLSTM) with Additive Attention and Dilated Convolution (BiLSTM-AADC) with Savitzky–Golay filtering. The methodology was applied to analyze data collected from a solar power probe located in Douala, Cameroon, including predicting horizons of 1 to 15 days. Solar power generation for 10 min to 180 min ahead was predicted by Rahman et al. [15] based on univariate and multivariate models using historical weather data, using the LSTM technique. The LSTM with Temporal Convolutional Network (TCN) model (LSTM-TCN) for forecasting PV energy was proposed by Limouni et al. [16], incorporating different seasons, cloudy, clear, and intermittent days.

Recently, Gupta et al. [17] presented a review of existing literature on ML techniques for PV forecasting, exploring various categories, including DL, hybrid, and mathematical models. Cao et al. [18] proposed a PV power prediction model based on the Stacking Ensemble Algorithm (StEA) coupled with correlation-guided fast Fourier transform to decompose the high- and low-frequency components. The model was tested on 12 PV data sets, and the results demonstrated good prediction accuracy. The amount of energy generated by solar cells employing ML techniques such as Random Forest (RF), k-Nearest Neighbors (k-NN), Extra Tree (ET), and XGBoost was evaluated using various statistical parameters in [19].

In the last decade, Sun et al. [20] used a CNN on the SUNSET dataset to forecast solar PV power 15 min ahead. This shows that cloud movement introduces significant uncertainty in short-term solar power forecasting. Another CNN model called SolarNet was proposed by Korkmaz [21] for short-term PV output power prediction under different weather conditions and seasons, compared with different DL models. Using LSTM networks, the influence of weather conditions was introduced for forecasting PV time series [22]. An additional study has also employed LSTM, Jung et al. [23] evaluated space–time data, using data from 63 months and 164 locations, using LSTM to propose new locations for the implementation of PV systems.

Table 1 provides a summary of some studies of solar PV forecasting techniques using AI-based, ensemble, and hybrid models. It is worth noting that this review does not encompass all studies in the area, as it is not a review paper. Most existing models focus on improving short-term prediction accuracy but often overlook interpretability, making it challenging for operators to understand the impact of individual meteorological and operational variables on the forecasts. Additionally, while hybrid models combining different techniques have shown potential, their integration strategies are still limited, lacking efficient mechanisms to exploit complementary strengths fully. Another gap lies in the treatment of uncertainty: many forecasting studies provide point estimates without incorporating multi-scenario frameworks capable of supporting operational decision-making under varying conditions. To address this gap, this study focuses on key research questions in this domain.

- How can forecasting models be designed to enhance interpretability and provide clear insights into the impact of individual meteorological and operational variables on PV energy predictions?
- What integration strategies can be developed for hybrid models to efficiently exploit the complementary strengths of different forecasting techniques and improve overall prediction performance?
- How can multi-scenario forecasting frameworks be incorporated into PV energy prediction models to better address uncertainty and support operational decision-making under varying conditions?

To address these questions, this study proposes a novel hybrid forecasting framework for multi-hour-ahead solar PV generation prediction. The hybrid model FNO-BiLSTM-NAM incorporates three domain-specific neural components: A Fourier Neural Operator (FNO), BiLSTM, and a Neural Additive Model (NAM). The study employs preprocessing steps, including anomaly detection, imputation, and feature engineering. The framework dynamically selects input variables, such as global radiation, air and PV panel temperature, wind speed, meteorological, and temporal. These components enable accurate, interpretable, and robust forecasting, as validated through independent experiments on a PV dataset for prediction horizons ranging from 1 to 96 h ahead.

The contributions of this study are:

- The paper proposes a hybrid forecasting framework that enhances interpretability by integrating an NAM model, allowing explicit quantification of the contribution of individual meteorological and operational variables to PV energy predictions.
- It introduces an optimized integration strategy that combines FNO for spectral–temporal feature extraction, BiLSTM networks for sequential dependency modeling, and NAM for feature-wise interpretability, effectively exploiting the complementary strengths of different forecasting techniques.
- The study implements a multi-scenario forecasting framework based on low, average, and high uncertainty levels, enabling operators to make more informed operational decisions and improve risk management in PV power generation.

The remainder of this paper is organized as follows: Section 1 reviews related work on time series prediction models, with a focus on PV energy forecasting using ML and deep learning techniques. Section 2 introduces the new dataset, providing an overview of its main characteristics, including exploratory data analysis, anomaly detection procedures, and an assessment of feature importance. Section 3 describes the overall architecture of the proposed hybrid model, detailing its core components and integration strategy. Section 4 presents the experimental evaluation, including performance comparisons against

Table 1
Review of forecasting models explored for PV power forecasting.

Reference	Models	Sampling time	Period	Forecasting horizon
[12]	ANNs	1 h	1 year	1, 2, 3 h
[11]	ANNs	10 min	3 years	10 min
[8]	GRU	5 min	4 months	–
[20]	CNN	10 s, 2 min	3,000 h	15 min
[21]	CNN	1 h	2 years	1, 2, 3 h
[22]	LSTM	15 min	1 year	1 h
[23]	LSTM	1 h	63 months	1–12 months
[15]	LSTM	5 min	1 year	10–180 min
[7]	Ensemble	15 min	years	15 min, 1 h, 1 day
[9]	Ensemble	1 h	15 months	24, 48, 72 h
[18]	Ensemble	15 min	1 year	1–24 h
[19]	Ensemble	5 min	1 year	–
[6]	ANN and DNN	15 min	1 year	7 days
[10]	LSTM and CNN	10 min	3 years	10 min
[16]	LSTM and TCN	5 min	3 years	15–30 min
[5]	SARIMA and RVFL	20 min	4 months	20, 40, 60 min
[14]	BiLSTM and AADC	30 min	1 year	1, 5, 7, 15 days
[13]	ISt, LSTM, and Informer	1 h	4 years	1–336 h

benchmark models and an analysis of forecasting accuracy under different scenarios. Finally, Section 5 summarizes the key findings and outlines potential directions for future research.

2. Materials

This section presents an overview of the main characteristics of the dataset, the exploratory data analysis, and the anomaly detection procedure. Together, these components establish a systematic, data-driven foundation for developing reliable and effective time series forecasting models.

2.1. Dataset description

To evaluate the superiority and reliability of the hybrid FNO-BiLSTM-NAM model, a short-term forecasting study was conducted with an original time-series dataset of solar power generation. The dataset was collected from PLIN Energy, a company located in Loanda, in southern Brazil. The plant is located at −22.90835 latitude and −53.112785 longitude, with an altitude of 495 m above sea level. The plant consists of five independent lots, each with a capacity of 1 MW. PLIN Energy operates in the Brazilian distributed generation market, offering discounts on the electricity bill to low-voltage consumers of the utility.

The plant has a capacity installed of 5 MW of inverter power (6.65 MWp) and is composed of 16,400 solar panels of 400 W each. The climate of Loanda has a hot season that lasts approximately six months, from October to April, with temperatures frequently above 31 °C, peaking in February (maximum of 32 °C and minimum of 22 °C). The cold season lasts about three months, from May to July, with temperatures below 26 °C, with June being the coldest month (maximum of 25 °C and minimum of 15 °C). Fig. 1 presents the PV plant layout. In this PV farm, there are 5 distinct modules, B, C, D, E, and F, composed of various PV cells. The module B data will be used in this study.

The dataset provided by this PV system includes the PV energy (kWh) output and the corresponding climate data, input, being recorded: global radiation (W/m^2), air temperature (°C), panel temperature (°C), and wind speed (km/h). Table 2 presents the descriptive statistics of the Minimum (Min), Maximum (Max), mean, Standard deviation (Std), kurtosis, and skewness of the dataset. The same table descriptive statistics for the PV solar from the year 2022, sampled hourly from 5 am to 7 pm, reveal significant variability in key environmental and operational parameters. This study considers energy units in kWh since our dataset provides measurements aggregated over fixed time intervals.

Global radiation exhibits a wide range with a high standard deviation, indicating fluctuating solar input. Air and panel temperatures reach extreme values, with high kurtosis and positive skewness, suggesting occasional extreme heating, potentially due to sensor anomalies or localized conditions. Wind speeds remain consistently low with minimal variation, while PV power generation varies across modules and has moderate skewness. Negative kurtosis in global radiation and PV energy suggests flatter distributions. These findings highlight the dynamic nature of solar power systems and the potential impact of environmental fluctuations on performance.

The Open-Meteo was used to obtain two variables, used with input, cloud cover, and precipitation data. It is an open-source weather API (Application Programming Interface) and offers free access for use, available at https://open-meteo.com/. Raw meteorological data were used to generate new exogenous features by incorporating temporal features extracted directly from the timestamp index. The forecasting also focused on three key temporal indicators: hour ($\in \{0, 1, 2, \ldots, 23\}$), day of the week ($\in \{0, 1, \ldots, 6\}$), and month ($\in \{1, 2, \ldots, 12\}$). These categorical temporal features were derived from the date-time index of the time series data and provided to the model as separate input channels. This approach enables the model to learn distinct patterns associated with different hours of the day, days of the week, and months of the year without requiring explicit cyclic encoding.

2.2. Dataset preprocessing

The preprocessing step ensures that the collected data is suitable as input for the selected models. Initially, filtering removes time periods when no sunlight data is available, and the period used was 5 am to 7 pm. The preprocessing stage prepares the data for encoding by performing feature scaling, applying data transformations, partitioning datasets, and addressing missing values and negative data.

2.2.1. Dataset cleaning

The overall methodology to clean anomalies is detailed in Algorithm 1. Negative values were presented in certain timestamps within the raw dataset, indicating potential outliers. Although such values may occur in the data collection process, they lack practical significance in solar energy production. To improve the quality of the dataset, outliers were identified and removed, while missing values were estimated.

The quartiles are used to describe the distribution of a dataset by dividing it into four equal parts. The low scenario, represented by the first quartile (q25), indicates that 25% of the data values fall below this threshold, capturing lower-end variations or conservative outcomes. The average scenario, represented by the second quartile or median (q50), reflects the midpoint of the data, where half of the values are

Fig. 1. PV plant layout with five modules, located at Loanda, Parana, Brazil.

Table 2
Descriptive statistics of the dataset summarized with hourly sampling from 5 am to 7 pm.

Variable	Min	Max	Mean	Std	Kurtosis	Skewness
Global radiation (W/m^2)	0.05	1189.65	362.43	315.29	−1.21	0.39
Air temperature (°C)	7.93	38.78	25.02	5.69	−0.54	−0.17
Panel temperature (°C)	3.00	69.78	33.36	13.64	−0.89	0.32
Wind speed (km/h)	0.02	6.03	2.55	1.17	−0.48	0.32
PV energy (kWh)	6.8×10^0	4.9×10^5	1.91×10^5	1.28×10^5	−1.11	0.28

below and half are above, serving as a central tendency indicator. The high scenario, represented by the third quartile (q75), signifies that 75% of the data values fall below this point, capturing upper-end variations or optimistic outcomes.

Fig. 2 illustrates a comparison of the PV energy (kWh) data before and after the preprocessing procedure. In this figure, high-frequency peaks are disregarded, as these fluctuations are not representative of energy generation. After this filtering, the data is selected according to its results for the model, as explained in the next sub-section.

2.2.2. Dataset normalization

The dataset used in this study spans from December 21, 2021, to December 20, 2022, with data points collected during daylight hours (5 to 19). We applied Z-score normalization [24] to transform the output data so that they are on a common scale, without distorting the differences in the range of values. For model development and evaluation, the chronologically ordered data was divided into three distinct sets, training set (70% of data), validation set (10% of data), and test set (20% of data), i.e., the specific time periods for each set are, training, December 21, 2021 to September 26, 2022, validation, September 27, 2022 to October 30, 2022, testing, October 31, 2022 to December 20, 2022.

For each dataset, a sliding window approach was implemented to create input–output pairs. For each position in the dataset, a window of length `seq_len` was used as input, and the subsequent `pred_len` time steps were used as the prediction target. Fig. 3 illustrates the sliding window methodology employed in this study, with splitting data into training, validation, and test sets.

3. Methodology

In this section, the proposed FNO-BiLSTM-NAM that generates probabilistic predictions is described. This approach addresses the challenges of PV energy forecasting: non-linear patterns, multi-scale temporal dependencies, and uncertainty quantification. To comprehensively evaluate its performance, it will be compared with state-of-the-art models described in this section.

3.1. Summary of the proposed framework

Fig. 4 illustrates the comprehensive workflow of the proposed methodology that consists of four sequential steps: (i) data input, (ii) data preprocessing, (iii) application of the optimized FNO-BiLSTM-NAM (see Section 3.2 for model architecture and Section 3.3 for optimization strategy), and (iv) comparison of the results (compared AI-based forecasting models as presented in Section 3.4).

In the first step, data from the Loanda PV Farm (Brazil), weather data (cloud cover and precipitation), and temporal indicators (hour, day of the week, and month of the year) are used. In the second step, the preprocessing stage involves data cleaning with anomaly detection, handling missing values, and data normalization. After data splitting, feature engineering using SHapley Additive exPlanations (SHAP) and applying autocorrelation, AutoCorrelation Function (ACF), and Partial AutoCorrelation Function (PACF) functions are considered.

In the third step, the PV dataset was sampled at an hourly resolution. The predictive performance of multiple models was evaluated with hyperparameters optimized by the Bayesian technique, ML (RF, XGBoost, Light Gradient-Boosting Machine (LGBM), and Natural Gradient Boosting (NGBoost)) and DL (Tabular Prior-Data Fitted Network (TabPFN)), across various forecasting horizons, 1 h, 2 h, 6 h, 12 h, 24 h, 48 h, 72 h, and 96 h, and lag window sizes, 6, 12, and 20. The model performance and stopping criterion were measured using Mean Squared Error (MSE). In the last step, the main results are obtained and compared. The FNO-BiLSTM-NAM model is evaluated through multiple performance metrics and multi-scenario forecasting to provide robust predictions for operational use.

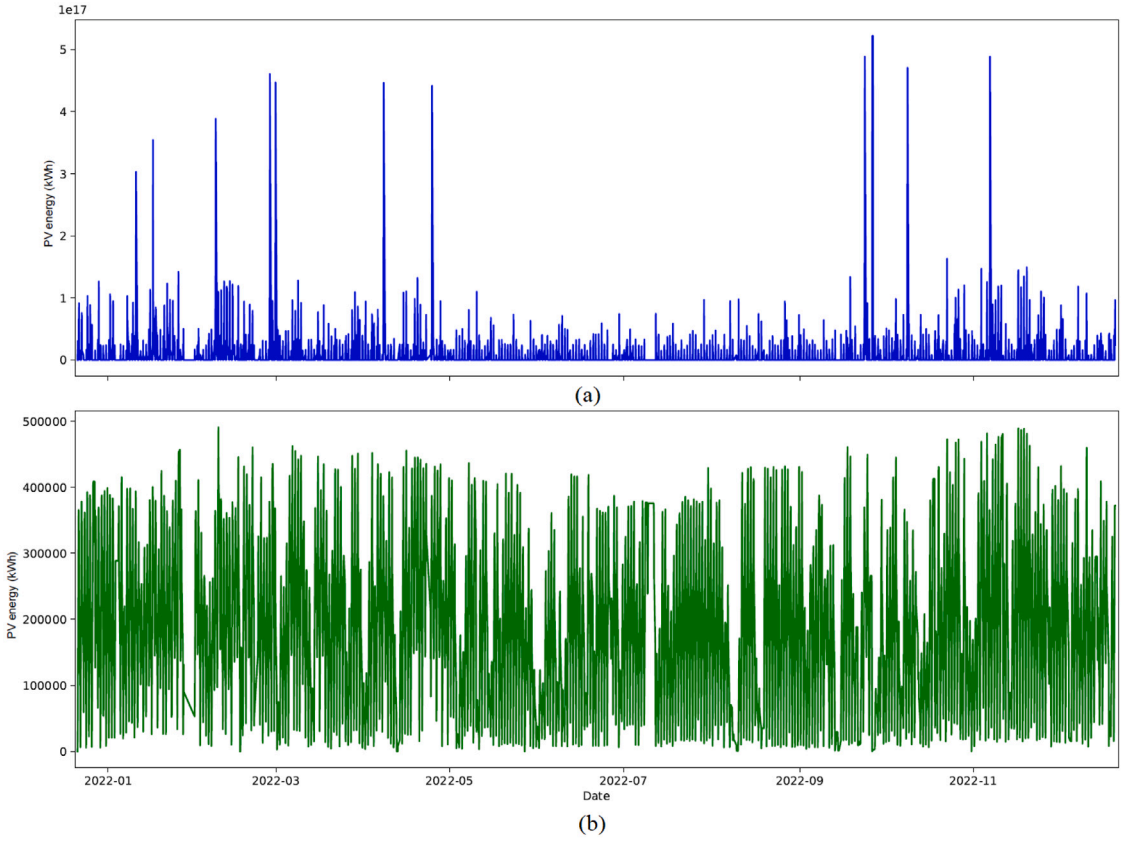

Fig. 2. PV energy (kWh) data profiles for the year 2022 with hourly sampling from 5 am to 7 pm for module B (a) before and (b) after the cleaning (procedure described in Algorithm 1).

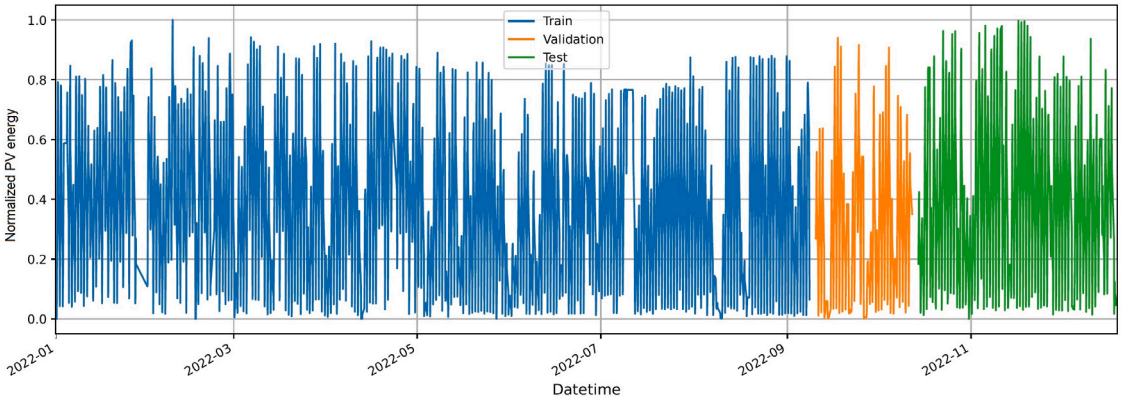

Fig. 3. Normalized PV energy data and splitting into training (70%), validation (10%), and test (20%) sets, sequence generated by the sliding window procedure.

3.2. FNO-BiLSTM-NAM architecture

The hybrid model combines three distinct neural network modules: FNO that processes input sequences in the frequency domain to capture long-range temporal patterns, a BiLSTM that models forward and backward temporal dependencies, and a NAM that provides feature-wise contributions for interpretability.

Fig. 5 presents the architecture of the proposed FNO-BiLSTM-NAM. Initially, the input features are processed through a feature attention mechanism, enabling the model to focus on the most relevant information. These refined features are then passed to a one-dimensional FNO (FNO1D) that extracts features (global temporal patterns) of the time series as explained in Section 3.2.1. The FNO1D is followed by a time embedding layer that encodes sequential dependencies. The encoded representations are fed into stacked BiLSTM layers equipped with attention mechanisms to capture both forward and backward temporal dynamics. The BiLSTM has teacher forcing to guide the model as explained in 3.2.2. On the other pathway, the NAM is used to enhance the prediction based on the data features as presented in Section 3.2.3. The NAM block improves interpretability and captures non-linear dependencies.

During the decoding phase, teacher forcing is applied to stabilize the learning process by integrating actual observations into the prediction pipeline. The final outputs are generated through a fully connected output layer, and a weighted combination strategy is used to aggregate multiple predicted outputs effectively. By combining additive modeling, operator learning, temporal embeddings, attention, and recurrent

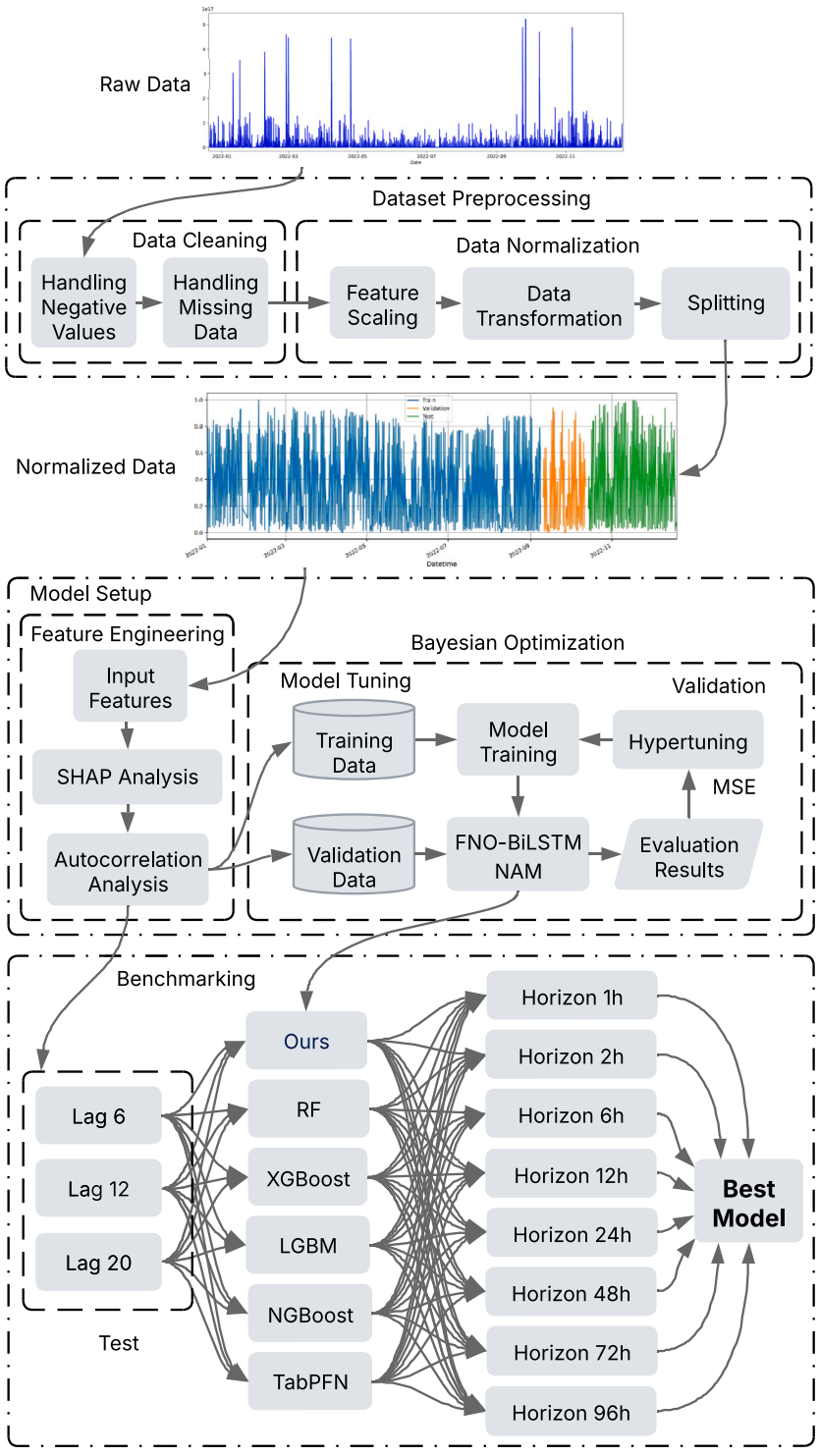

Fig. 4. Workflow of the structure of the proposed framework and its use in PV energy forecasting analysis.

sequence modeling, the proposed framework aims to achieve superior accuracy and robustness in temporal forecasting tasks.

The proposed FNO-BiLSTM-NAM model is trained considering a combination of loss functions (quantile loss, derivative loss, and peak penalty) as explained in Section 3.2.4. Finally, the exogenous variables used in the FNO-BiLSTM-NAM model are explained in Section 3.2.5.

Let $\mathbf{X} \in \mathbb{R}^{B\times T\times F}$ represent the input tensor, where B is the batch size, T is the temporal sequence length, and F is the number of input features. The model processes this input through a pipeline incorporating multiple techniques to extract and model relevant patterns. The model begins with a feature-wise attention mechanism $\mathbf{A} \in \mathbb{R}^{B\times F}$ that adaptively weights each input feature:

$$\mathbf{A} = \sigma(W_2 \cdot \mathrm{GELU}(W_1 \cdot \overline{\mathbf{X}})) \tag{1}$$

where $\overline{\mathbf{X}} \in \mathbb{R}^{B\times F}$ is the temporal average of input features, $W_1 \in \mathbb{R}^{F\times H}$ and $W_2 \in \mathbb{R}^{H\times F}$ are learnable parameters, H is the hidden dimension size, σ represents the sigmoid activation, and the Gaussian Error Linear

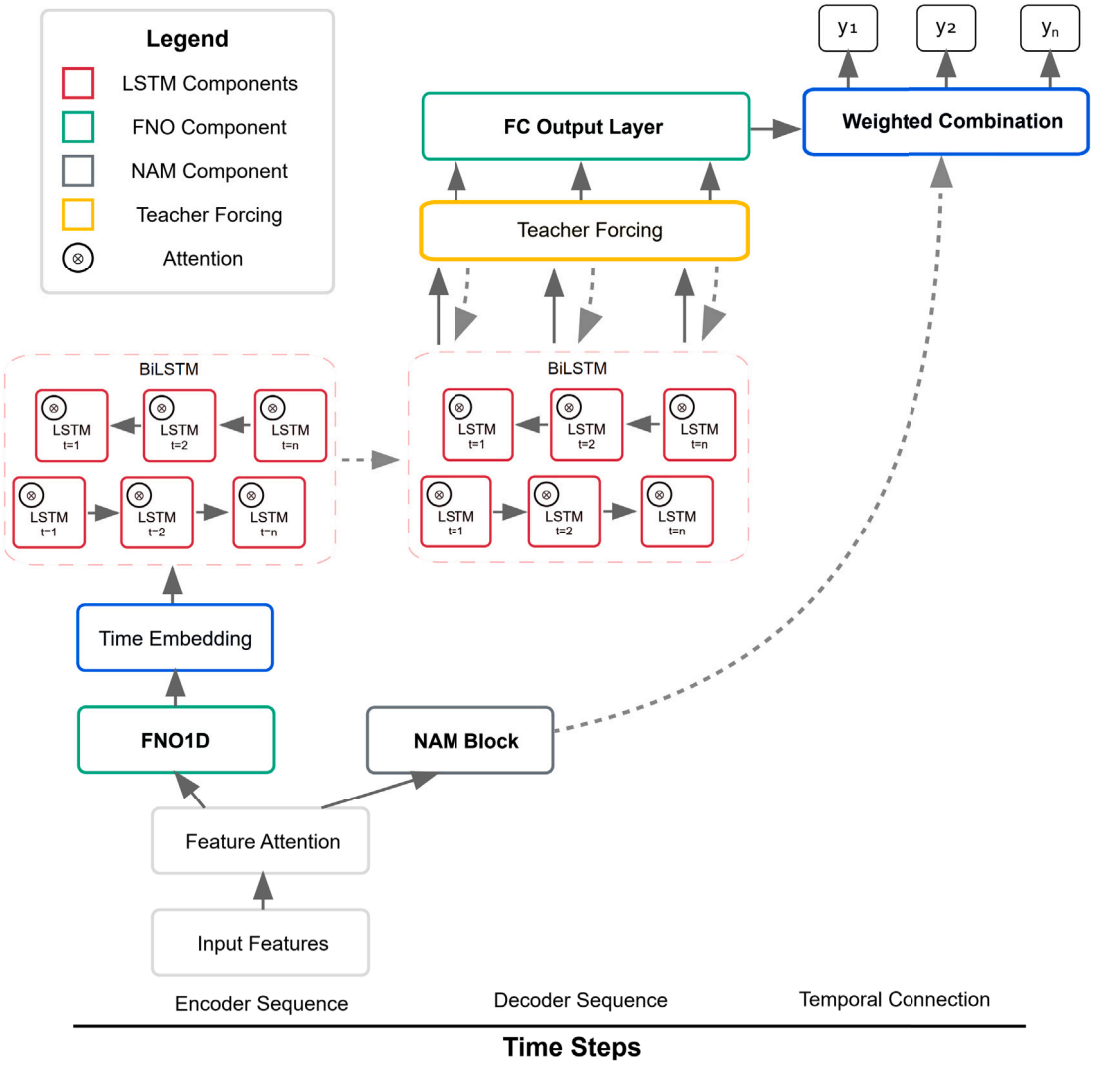

Fig. 5. Proposed FNO-BiLSTM-NAM architecture.

Unit (GELU) function is applied [25]. The attended features $\mathbf{X}_{\text{weighted}} \in \mathbb{R}^{B\times T\times F}$ are computed through element-wise multiplication:

$$\mathbf{X}_{\text{weighted}} = \mathbf{X} \odot \mathbf{A}. \tag{2}$$

This mechanism allows the model to focus on the most relevant features for prediction.

3.2.1. Fourier Neural Operator

The weighted features then flow through an FNO pathway that exploits the spectral properties of time series. This component operates in the frequency domain to capture multi-scale temporal patterns and long-range dependencies [26]. The FNO processing begins with a linear projection:

$$\mathbf{X}_{\text{proj}} = W_{\text{in}}\mathbf{X}_{\text{weighted}} + b_{\text{in}} \tag{3}$$

where $W_{\text{in}} \in \mathbb{R}^{C\times F}$ is the input projection weight matrix, $b_{\text{in}} \in \mathbb{R}^{C}$ is the bias vector, and C is the channel dimension of the projected features.

The projected features $\mathbf{X}_{\text{proj}} \in \mathbb{R}^{B\times T\times C}$ undergo a Fourier transform along the temporal dimension:

$$\mathbf{X}_{\text{ft}} = \mathcal{F}(\mathbf{X}_{\text{proj}}) \tag{4}$$

where $\mathcal{F}$ denotes the real Fast Fourier Transform (FFT) Yu et al. [27] and $\mathbf{X}_{\text{ft}} \in \mathbb{C}^{B\times C\times M}$ is the frequency domain representation with M frequency modes. In the frequency domain, a spectral convolution is performed using complex-valued learnable weights $\mathbf{W} \in \mathbb{C}^{O\times C\times M}$ through a tensor contraction:

$$\mathbf{X}^{\text{out}}_{\text{ft}}(b,o,m) = \sum_{c=1}^{C} \mathbf{X}_{\text{ft}}(B,C,M)\mathbf{W}(O,C,M) \tag{5}$$

where O is the output channel dimension, C represents the input channel index, and M is the frequency mode index.

An inverse Fourier transform returns the data to the temporal domain [28], followed by another linear transformation and activation:

$$\mathbf{X}_{\text{fno}} = \text{GELU}(W_{\text{out}}\mathcal{F}^{-1}(\mathbf{X}^{\text{out}}_{\text{ft}}) + b_{\text{out}}) \tag{6}$$

where $W_{\text{out}} \in \mathbb{R}^{F\times O}$ and $b_{\text{out}} \in \mathbb{R}^{F}$ are the output projection parameters, and $\mathbf{X}_{\text{fno}} \in \mathbb{R}^{B\times T\times F}$ is the output of the FNO module.

Temporal embedding $\mathbf{E}_{\text{time}} \in \mathbb{R}^{T\times F}$ is incorporated for solar energy forecasting due to its diurnal and seasonal patterns through dedicated embeddings:

$$\mathbf{E}_{\text{time}} = \mathbf{E}_{\text{hour}}(h) + \mathbf{E}_{\text{day}}(d) + \mathbf{E}_{\text{month}}(m) + \mathbf{E}_{\text{pos}} \tag{7}$$

where $\mathbf{E}_{\text{hour}} \in \mathbb{R}^{24\times F}$, $\mathbf{E}_{\text{day}} \in \mathbb{R}^{7\times F}$, and $\mathbf{E}_{\text{month}} \in \mathbb{R}^{12\times F}$ are learnable embedding matrices, $h \in \{0,1,\ldots,23\}$ represents the hour of day, $d \in \{0,1,\ldots,6\}$ represents the day of week, $m \in \{0,1,\ldots,11\}$ represents the month of year, and $\mathbf{E}_{\text{pos}} \in \mathbb{R}^{T\times F}$ is a learned positional embedding that encodes sequence order. This temporal embedding is combined

Algorithm 1: Data anomaly detection and cleaning applied to PV energy dataset.

Data: Time series $X = \{x_1, x_2, \ldots, x_n\}$
Result: Cleaned time series $\hat{X}$
Function `CleanSolarData(`X`)`:
  $X^{(1)} \leftarrow$ `RemoveExtremeOutliers(`X`)`;
  $X^{(2)} \leftarrow$ `DetectContextualAnomalies(`$X^{(1)}$`)`;
  $\hat{X} \leftarrow$ `ApplyFinalSmoothing(`$X^{(2)}$`)`;
  return $\hat{X}$;
Function `RemoveExtremeOutliers(`X`)`:
  $Q_1 \leftarrow \text{Quantile}(X, 0.25)$;
  $Q_3 \leftarrow \text{Quantile}(X, 0.75)$;
  $IQR \leftarrow Q_3 - Q_1$;
  $L \leftarrow Q_1 - 2 \cdot IQR$;
  $U \leftarrow Q_3 + 2 \cdot IQR$;
  $A_{iqr} \leftarrow \{i \in \{1, \ldots, n\} : x_i < L \vee x_i > U\}$; `// Anomaly indices`
  for $i \in A_{iqr}$ **do**
    $x_i \leftarrow \text{NaN}$; `// Replace anomalies with NaN`
  end
  $X' \leftarrow$ `TimeInterpolate(`X`)` ; `// Interpolate NaN values`
  return X';
Function `DetectContextualAnomalies(`X`)`:
  for $i \in \{1, \ldots, n\}$ **do**
    $\mu_i^{24} \leftarrow \frac{1}{24} \sum_{j=i-23}^{i} x_j$; `// Rolling mean with window=24`
    $\sigma_i^{24} \leftarrow \sqrt{\frac{1}{24} \sum_{j=i-23}^{i} (x_j - \mu_i^{24})^2}$; `// Rolling std`
    $d_i \leftarrow \frac{|x_i - \mu_i^{24}|}{\sigma_i^{24}}$; `// Deviation`
  end
  $A_c \leftarrow \{i \in \{1, \ldots, n\} : d_i > 2.5\}$; `// Contextual anomaly indices`
  for $i \in A_c$ **do**
    $x_i \leftarrow \text{NaN}$; `// Replace anomalies with NaN`
  end
  $X' \leftarrow$ `TimeInterpolate(`X`)` ; `// Interpolate NaN values`
  return X';
Function `ApplyFinalSmoothing(`X`)`:
  for $i \in \{2, \ldots, n-1\}$ **do**
    $\hat{x}_i \leftarrow \frac{1}{3}(x_{i-1} + x_i + x_{i+1})$; `// Centered rolling mean`
  end
  return $\hat{X}$;
Function `TimeInterpolate(`X`)`:
  $X_{interp} \leftarrow \text{LinearInterpolation}(X)$; `// Fill NaNs with linear interpolation`
  $X_{complete} \leftarrow \text{ForwardFill}(X_{interp})$; `// Fill remaining NaNs at edges`
  $X_{complete} \leftarrow \text{BackwardFill}(X_{complete})$; `// Fill any remaining NaNs`
  return $X_{complete}$;

with the FNO output $\mathbf{X}_{\text{fno}}$:

$$\mathbf{X}_{\text{combined}} = \mathbf{X}_{\text{fno}} + \mathbf{E}_{\text{time}}. \tag{8}$$

3.2.2. Bidirectional long short-term memory

The overall architecture follows a sequence-to-sequence (Seq2Seq) paradigm, which is particularly effective for time series forecasting tasks. In this framework, an encoder maps the input sequence $\mathbf{X}_{1:T}$ to a latent representation $\mathbf{H}_{\text{enc}}$, and a decoder subsequently generates the output prediction sequence $\mathbf{Y}_{1:P}$ conditioned on this representation.

The Seq2Seq design is advantageous in capturing temporal dependencies of varying lengths, as it allows the encoder to summarize the historical context into a compact state while the decoder progressively unfolds future predictions. This paradigm is widely adopted in both natural language processing and time series domains, owing to its ability to handle variable-length inputs and outputs and to accommodate autoregressive decoding strategies.

The sequence modeling component employs a BiLSTM [29] encoder that processes the temporally-enriched FNO output:

$$\mathbf{H}_{\text{enc}}, (\mathbf{h}_n, \mathbf{c}_n) = \text{LSTM}_{\text{enc}}(\mathbf{X}_{\text{combined}}) \tag{9}$$

where $\mathbf{H}_{\text{enc}} \in \mathbb{R}^{B \times T \times 2D}$ is the encoder hidden state sequence, $\mathbf{h}_n \in \mathbb{R}^{B \times 2D}$ is the final hidden state, $\mathbf{c}_n \in \mathbb{R}^{B \times 2D}$ is the final cell state, and D is the LSTM hidden dimension.

For the encoder, forward $\mathbf{h}_n^{\text{forward}} \in \mathbb{R}^{B \times D}$ and backward $\mathbf{h}_n^{\text{backward}} \in \mathbb{R}^{B \times D}$ states are combined $\mathbf{h}_n^{\text{combined}} \in \mathbb{R}^{B \times D}$ through a linear projection:

$$\mathbf{h}_n^{\text{combined}} = \text{LinearProj}([\mathbf{h}_n^{\text{forward}}; \mathbf{h}_n^{\text{backward}}]) \tag{10}$$

where $[\cdot;\cdot]$ denotes concatenation along the feature dimension.

This processing incorporates both past and future within the input sequence [30]. The decoder employs an adaptive teacher forcing mechanism during training, with probability $\tau \in [0, 1]$, the model uses ground truth values as input rather than its predictions:

$$\mathbf{y}_{t+1} = \begin{cases} \text{LSTM}_{\text{dec}}(\mathbf{y}_t^{\text{true}}, \mathbf{h}_t, \mathbf{c}_t), & \text{with probability } \tau \\ \text{LSTM}_{\text{dec}}(\mathbf{y}_t^{\text{pred}}, \mathbf{h}_t, \mathbf{c}_t), & \text{with probability } 1 - \tau \end{cases} \tag{11}$$

where $\mathbf{y}_t^{\text{true}}$ is the ground truth at time t, $\mathbf{y}_t^{\text{pred}}$ is the model's prediction at time t, and $\mathbf{h}_t \in \mathbb{R}^{B \times D}$ and $\mathbf{c}_t \in \mathbb{R}^{B \times D}$ are the hidden and cell states at time t, respectively. The teacher-forcing ratio decays over time as $\tau_t = \tau_0 \cdot \gamma^t$ where $\tau_0 \in [0, 1]$ is the initial teacher-forcing probability, $\gamma \in [0, 1]$ is the decay rate, and t is the training epoch.

This approach helps bridge the gap between training and inference modes. The BiLSTM decoder outputs are projected to the desired quantiles through a Fully Connected (FC) feed-forward network:

$$\mathbf{X}_{\text{LSTM}} = \text{FC}_{\text{out}}(\mathbf{H}_{\text{dec}}) \tag{12}$$

where $\mathbf{H}_{\text{dec}} \in \mathbb{R}^{B \times P \times D}$ is the decoder's output sequence for P prediction horizons, and $\mathbf{X}_{\text{LSTM}} \in \mathbb{R}^{B \times P \times Q}$ is the LSTM pathway's output for Q quantiles.

3.2.3. Neural Additive Model

Parallel to the sequence modeling path, NAM was employed, which independently processes each input feature:

$$h_f(\mathbf{X}_f) = W_{f,2}\text{GELU}(W_{f,1}\mathbf{X}_f + b_{f,1}) + b_{f,2} \tag{13}$$

where $\mathbf{X}_f \in \mathbb{R}^{B \times T}$ is the fth feature from the input tensor, $W_{f,1}$, $W_{f,2}$, $b_{f,1}$, and $b_{f,2}$ are feature-specific learnable parameters, and $h_f(\mathbf{X}_f) \in \mathbb{R}^{B \times P \times Q}$ is the contribution of feature f to the forecasts.

Each feature-specific network generates contributions across all prediction horizons and quantiles, which are then weighted and summed, such as:

$$\mathbf{X}_{\text{NAM}} = \sum_{f=1}^{F} \sigma(w_f) \cdot h_f(\mathbf{X}_f) \tag{14}$$

where $w_f \in \mathbb{R}$ are learnable feature weights. This additive structure enables the analysis of each feature's contribution to the final prediction and combines modeling approaches using learned weights:

$$\hat{\mathbf{Y}} = \alpha \cdot \mathbf{X}_{\text{LSTM}} + \beta \cdot \mathbf{X}_{\text{NAM}} \tag{15}$$

where $[\alpha, \beta] = \text{softmax}([w_{\text{LSTM}}, w_{\text{NAM}}])$ ensures proper weighting with $w_{\text{LSTM}} \in \mathbb{R}$ and $w_{\text{NAM}} \in \mathbb{R}$ being learnable parameters. This integration leverages the temporal modeling capabilities of the LSTM and the feature-specific modeling of the NAM. The output $\hat{\mathbf{Y}} \in \mathbb{R}^{B \times P \times Q}$ provides probabilistic forecasts with P prediction horizons and Q quantiles (typically $Q = 3$ for the 25th, 50th, and 75th percentiles).

3.2.4. Loss function and training

The model is trained using a total loss function with three key objectives: quantile loss for calibrated probabilistic forecasts [31], derivative loss to preserve temporal dynamics, and peak penalty to prevent underestimation of production peaks,

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{quantile}} + \lambda_1 \cdot \mathcal{L}_{\text{derivative}} + \lambda_2 \cdot \mathcal{L}_{\text{peak}} \tag{16}$$

$$\mathcal{L}_{\text{quantile}} = \sum_{q\in\{0.25,0.5,0.75\}} w_q \cdot \mathbb{E}[\max((q-1)\cdot e, q\cdot e)] \tag{17}$$

$$\mathcal{L}_{\text{derivative}} = \mathbb{E}[w_t \cdot ((\hat{y}_{t+1} - \hat{y}_t) - (y_{t+1} - y_t))^2] \tag{18}$$

$$\mathcal{L}_{\text{peak}} = \mathbb{E}[\max(0, y - \hat{y}_{0.5})^2] \tag{19}$$

where $e = y - \hat{y}_q$ is the prediction error for quantile q, $y \in \mathbb{R}^{B\times P}$ represents the ground truth values, $\hat{y}_q \in \mathbb{R}^{B\times P}$ is the prediction for quantile q, $w_q \in \mathbb{R}^+$ are the weights for each quantile, and $w_t \in \mathbb{R}^+$ are time-decaying weights that emphasize near-future consistency. The hyperparameters $\lambda_1, \lambda_2 \in \mathbb{R}^+$ control the relative importance of each loss component. The Algorithm 2 presents the training procedure of the proposed model.

Algorithm 2: Training–validation procedure of the FNO-BiLSTM-NAM.

Input: Time series data $\mathbf{X} \in \mathbb{R}^{B\times T\times F}$, Ground truth targets $\mathbf{y} \in \mathbb{R}^{B\times P}$, Time features (hour h, day d, month m), Hyperparameters: FNO modes M, hidden dims H, Teacher forcing initial ratio τ_0, decay rate γ

Output: Trained model, Probabilistic forecasts $\hat{\mathbf{Y}} \in \mathbb{R}^{B\times P\times Q}$

```
// Model Training Procedure
Initialize model parameters
Define optimizer (Adam)
τ ← τ_0 ;                                   // Initial teacher forcing ratio
for epoch ← 1 to max_epochs do
    Set the model to training mode for batch (X^(i), y^(i), h^(i), d^(i), m^(i)) in
     training_loader do
        Zero gradients
        Ŷ^(i) ← ForwardX^(i), h^(i), d^(i), m^(i), y^(i), τ ; // Forward with TF
        L_quantile ← Σ_{q∈{0.25,0.5,0.75}} w_q · E[max((q−1)·e, q·e)]
        L_derivative ← E[w_t · ((ŷ_{t+1} − ŷ_t) − (y_{t+1} − y_t))^2]
        L_peak ← E[max(0, y − ŷ_0.5)^2]
        L_total ← L_quantile + λ_1 · L_derivative + λ_2 · L_peak ; // Composite loss
        Compute gradients of L_total
        Apply gradient clipping: ‖∇‖ ≤ τ_clip ;  // Prevent exploding
         gradients
        Update parameters with optimizer
    end
    Set model to evaluation mode; L_val ← 0 ;    // Validation phase
    for batch (X^(j), y^(j), h^(j), d^(j), m^(j)) in validation_loader do
        Ŷ^(j) ← ForwardX^(j), h^(j), d^(j), m^(j) ;  // No teacher forcing
        Calculate L_total^(j) as before
        L_val ← L_val + L_total^(j)
    end
    Update learning rate based on L_val (ReduceLROnPlateau)
    if L_val improved then
        Save the best model; Reset the patience counter
    end
    else
        Increment patience counter if patience counter > max_patience
         then
            Break ;                          // Early stopping triggered
        end
    end
    τ ← τ_0 · γ^epoch ;                      // Decay teacher forcing ratio
end
Load best model weights
return Trained model
```

3.2.5. Exogenous variables

In time series forecasting, there are the target variable and exogenous variables that provide additional information for the target variable [32]. In the proposed model, during inference, multi-step forecasts are generated, and exogenous variables that may or may not be known in advance are incorporated. The target time series is $y = \{y_1, y_2, \ldots, y_T\}$ and the exogenous variables is $V = \{V_1, V_2, \ldots, V_T\}$, where each $V_t = \{v_{t,1}, v_{t,2}, \ldots, v_{t,K}\}$ represents K exogenous features at time t.

For an AutoRegressive (AR) model [33] with a prediction horizon of h steps and a look-back window of L, the forecasting objective can be expressed as:

$$\hat{y}_{t+1:t+h} = f(y_{t-L+1:t}, V_{t-L+1:t}, V_{t+1:t+h}) \tag{20}$$

where t is the time, $\hat{y}_{t+1:t+h}$ represents the predicted values for the next h time steps.

For exogenous variables, a scenario was used that represents a statistical pattern derived from historical data,

$$V_{t+i}^{(s)} = \phi_s(V_{t-L+1:t}, \theta_i) \tag{21}$$

where $V_{t+i}^{(s)}$ is the projected exogenous variable at time $t+i$ under scenario s, and ϕ_s is a scenario-specific projection function parameterized by θ_i, which may include time-of-day effects or other temporal factors. Three primary scenarios were defined, low scenario, $\theta_i = q_{0.25,i}$ (25th percentile for time i), average scenario: $\theta_i = \mu_i$ (mean value for time i), and high scenario, $\theta_i = q_{0.75,i}$ (75th percentile for time i).

The AR mechanism iteratively constructs the forecast:

$$\hat{y}_{t+i} = g(y_{t-L+1:t}, \hat{y}_{t+1:t+i-1}, V_{t-L+1:t}, V_{t+1:t+i}^{(s)}) \tag{22}$$

where $\hat{y}_{t+i}$ is the prediction for time $t+i$, and g is the model's prediction function.

In the proposed model, the NAM maintains a sliding window C_i that is updated at each prediction step:

$$C_0 = [y_{t-L+1:t}, V_{t-L+1:t}] \tag{23}$$

$$C_i = [C_{i-1,2:L}, [\hat{y}_{t+i-1}, V_{t+i}^{(s)}]] \tag{24}$$

where $C_{i,2:L}$ are the elements 2 to L of the window C_i, a sliding window that maintains the sequence length required for the neural components.

The NAM architecture combines specialized components for different aspects of the forecasting task. The forecast at each time step is formulated as:

$$\hat{y}_{t+i} = \alpha \cdot f_{LSTM}(C_i) + (1-\alpha)\cdot f_{NAM}(C_i) \tag{25}$$

where f_{LSTM} is the output from a sequence-to-sequence LSTM network, f_{NAM} is the contribution from a NAM that captures feature-specific patterns, and α is a learned weighting parameter.

The NAM processes each feature individually:

$$f_{NAM}(C_i) = \sum_{k=1}^{K+1} w_k \cdot f_k(C_{i,:,k}) \tag{26}$$

where f_k is a feature-specific network, $C_{i,:,k}$ is the sequence for feature k in window C_i, and w_k is a learned feature weight. This formulation allows the model to adapt to different exogenous variable scenarios while maintaining coherent autoregressive predictions across multiple time steps.

Algorithm 3 presents the auxiliary functions that are used in the FNO-BiLSTM-NAM model. The core of the proposed model is the neural network architecture, which integrates three specialized components: a FNO for spectral analysis, a BiLSTM for sequence modeling, and a NAM for feature-specific contributions. These components work to capture the complex temporal dynamics of solar power generation.

3.3. Hyperparameter tuning method

To determine the hyperparameter-optimized configuration for the teacher-forcing prediction model, Bayesian optimization via `Ray Tune` with the Asynchronous Successive Halving Algorithm (ASHA) scheduler [34] was used. The ASHA allocates computational resources by terminating poorly performing configurations early. It operates with validation loss ($\mathcal{L}_{\text{val}}$), minimization, maximum epochs per trial equals 5, 2 epochs of learning, and reduction factor equals 2. The hyperparameter tuning process follows:

$$\theta^* = \arg\min_{\theta\in\Theta} \mathcal{L}_{\text{val}}(f_\theta(\mathcal{D}_{\text{train}}), \mathcal{D}_{\text{val}}) \tag{27}$$

Algorithm 3: Auxiliary functions to FNO-BiLSTM-NAM.

Function `FeatureAttention(X)`:
- $\overline{\mathbf{X}} \leftarrow \text{Mean}(\mathbf{X}, \text{dim}=1)$; `// Temporal average`
- $\mathbf{A} \leftarrow \sigma(W_2 \cdot \text{GELU}(W_1 \cdot \overline{\mathbf{X}}))$; `// Compute attention weights`
- $\mathbf{X}_{\text{weighted}} \leftarrow \mathbf{X} \odot \mathbf{A}$; `// Apply attention`
- **return** $\mathbf{X}_{weighted}$, $\mathbf{A}$

Function `FNO1D(`$\mathbf{X}_{weighted}$`)`:
- $\mathbf{X}_{\text{proj}} \leftarrow W_{\text{in}}\mathbf{X}_{\text{weighted}} + b_{\text{in}}$; `// Input projection`
- $\mathbf{X}_{\text{ft}} \leftarrow \mathcal{F}(\mathbf{X}_{\text{proj}})$; `// Fourier transform`
- $\mathbf{X}_{\text{ft}}^{\text{out}} \leftarrow \text{TensorContract}(\mathbf{X}_{\text{ft}}, \mathbf{W})$; `// Spectral convolution`
- $\mathbf{X}_{\text{fno}} \leftarrow \text{GELU}(W_{\text{out}}\mathcal{F}^{-1}(\mathbf{X}_{\text{ft}}^{\text{out}}) + b_{\text{out}})$; `// Inverse transform`
- **return** $\mathbf{X}_{fno}$

Function `TimeEmbedding(`$\mathbf{X}_{fno}$, h, d, m`)`:
- $\mathbf{E}_{\text{hour}} \leftarrow \text{Embedding}(h)$; $\mathbf{E}_{\text{day}} \leftarrow \text{Embedding}(d)$; $\mathbf{E}_{\text{month}} \leftarrow \text{Embedding}(m)$
- $\mathbf{E}_{\text{time}} \leftarrow \mathbf{E}_{\text{hour}} + \mathbf{E}_{\text{day}} + \mathbf{E}_{\text{month}} + \mathbf{E}_{\text{pos}}$
- $\mathbf{X}_{\text{combined}} \leftarrow \mathbf{X}_{\text{fno}} + \mathbf{E}_{\text{time}}$
- **return** $\mathbf{X}_{combined}$

Function `NAMBlock(`$\mathbf{X}_{weighted}$`)`:
- **for** *each feature* f *in* F **do**
  - $\mathbf{X}_f \leftarrow \mathbf{X}_{\text{weighted}}[:, :, f]$; `// Extract feature`
  - $h_f(\mathbf{X}_f) \leftarrow W_{f,2}\text{GELU}(W_{f,1}\mathbf{X}_f + b_{f,1}) + b_{f,2}$; $h_f(\mathbf{X}_f) \leftarrow \text{Reshape}(h_f(\mathbf{X}_f), [B, P, Q])$
- **end**
- $\mathbf{X}_{\text{NAM}} \leftarrow \sum_{f=1}^{F} \sigma(w_f) \cdot h_f(\mathbf{X}_f)$; `// Weighted feature sum`
- **return** $\mathbf{X}_{NAM}$

Function `Forward(`$\mathbf{X}$, h, d, m, $\mathbf{y}^{true}$=*None*, τ=*None*`)`:
- $\mathbf{X}_{\text{weighted}}, \mathbf{A} \leftarrow$ `FeatureAttention(X)` ; `// Feature attention`
- $\mathbf{X}_{\text{fno}} \leftarrow$ `FNO1D(`$\mathbf{X}_{weighted}$`)` ; `// FNO path`
- $\mathbf{X}_{\text{combined}} \leftarrow$ `TimeEmbedding(`$\mathbf{X}_{fno}$, h, d, m`)`
- $\mathbf{H}_{\text{enc}}, (\mathbf{h}_n, \mathbf{c}_n) \leftarrow$ `BidirectionalLSTMEncoder(`$\mathbf{X}_{combined}$`)` ; `// BiLSTM Encoder`
- **if** *bidirectional* **then**
  - $\mathbf{h}_n^{\text{combined}} \leftarrow \text{LinearProj}([\mathbf{h}_n^{\text{forward}}; \mathbf{h}_n^{\text{backward}}])$; $\mathbf{c}_n^{\text{combined}} \leftarrow \text{LinearProj}([\mathbf{c}_n^{\text{forward}}; \mathbf{c}_n^{\text{backward}}])$
  - $\mathbf{h}_n \leftarrow \mathbf{h}_n^{\text{combined}}$; $\mathbf{c}_n \leftarrow \mathbf{c}_n^{\text{combined}}$
- **end**
- $\mathbf{y}_0 \leftarrow \text{ZeroTensor}(B, 1, Q)$; $\mathbf{H}_{\text{dec}} \leftarrow []$; `// Decoder init`
- **for** $t \leftarrow 0$ **to** $P-1$ **do**
  - **if** *training and* $\mathbf{y}^{true}$ *is not None and Random()* $< \tau$ **then**
    - $\mathbf{y}_{\text{input}} \leftarrow \mathbf{y}^{\text{true}}[:, t]$ if $t > 0$ else $\mathbf{y}_0$; `// Teacher forcing`
  - **else**
    - $\mathbf{y}_{\text{input}} \leftarrow \mathbf{y}_{\text{pred}}$ if $t > 0$ else $\mathbf{y}_0$; `// Own prediction`
  - **end**
  - $\mathbf{h}_t, \mathbf{c}_t, \mathbf{y}_{\text{pred}} \leftarrow$ `LSTMDecoder(`$\mathbf{y}_{input}$, $\mathbf{h}_n$, $\mathbf{c}_n$`)`
  - Append $\mathbf{y}_{\text{pred}}$ to $\mathbf{H}_{\text{dec}}$
- **end**
- $\mathbf{X}_{\text{LSTM}} \leftarrow \text{FC}_{\text{out}}(\mathbf{H}_{\text{dec}})$
- $\mathbf{X}_{\text{NAM}} \leftarrow$ `NAMBlock(`$\mathbf{X}_{weighted}$`)` ; `// NAM path`
- $[\alpha, \beta] \leftarrow \text{softmax}([w_{\text{LSTM}}, w_{\text{nam}}])$; `// Component weights`
- $\hat{\mathbf{Y}} \leftarrow \alpha \cdot \mathbf{X}_{\text{LSTM}} + \beta \cdot \mathbf{X}_{\text{NAM}}$; `// Final integration`
- **return** $\hat{\mathbf{Y}}$

where θ represents the hyperparameter configuration from the search space Θ, covering both architectural and training-specific parameters, f_θ is teacher-forcing neural network model trained with configuration θ on the training dataset D_{train}, and $\mathcal{L}_{\text{val}}$ is the validation loss evaluated on the validation dataset D_{val}.

The optimized hyperparameter to FNO-BiLSTM-NAM and its functional roles are described as follows:

- `initial_tf_ratio`: The initial probability of applying teacher forcing during sequence-to-sequence model training. The teacher forcing guides the model by feeding ground-truth previous tokens as input during training rather than predicted outputs.
- `tf_decay`: A decay factor that exponentially reduces initial_tf_ratio over training steps or epochs, gradually shifting the model from teacher-forced to autoregressive predictions.
- `fno_dim`: The number of Fourier modes retained in the frequency domain for spectral-based neural operators, controlling the resolution of frequency-space representations.
- `num_layers`: The depth of the neural network determines its capacity to learn hierarchical representations. While deeper networks (higher num_layers) can capture more complex features, they also increase the risk of overfitting and computational overhead.
- `hidden_dim`: The dimensionality of the hidden state directly influences model expressivity. Larger values enhance representational power but escalate memory and computational requirements.
- `dropout`: A regularization technique that randomly deactivates neurons during training with a determined probability, preventing overfitting by promoting robust feature learning.

3.4. Comparison setup

This study employs RF [35], XGBoost [36], LGBM [37], NGBoost, and TabPFN as the base models for comparison. The application of these ensemble-based and neural models is justified for time series forecasting due to their ability to capture complex, nonlinear dependencies within temporal data [38]. RF, XGBoost, and LGBM leverage decision-tree-based ensembles to effectively model intricate relationships and handle high-dimensional features while mitigating overfitting. NGBoost [39] extends these capabilities by incorporating probabilistic predictions, enabling the quantification of uncertainty, which is particularly relevant in time-dependent contexts.

TabPFN [40], on the other hand, exploits prior-data fitting and transformer-based architectures to generalize well even on small datasets, offering competitive accuracy without extensive hyperparameter tuning. Together, these models provide robust, flexible, and scalable solutions for forecasting tasks, making them suitable for diverse temporal patterns and varying data availability. The stability and performance of these models have been validated in previous studies [41]. Upon follow-up, a detailed description of these base models is provided.

The forecasting models are assessed using MSE, Root MSE (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). MSE, given by Eq. (28), computes the average squared difference between actual and predicted values [42]. RMSE, calculated by Eq. (29), is the square root of MSE, giving more weight to larger errors. MAE, given by Eq. (30), calculates the average absolute deviation, regardless of error direction. MAPE, calculated by Eq. (31), expresses errors as percentages, making results scale-independent and comparable across datasets [43].

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{28}$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{29}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{30}$$

$$\text{MAPE} = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{31}$$

where y_i represents the actual values, $\hat{y}_i$ is the predicted values by the models, n the number of observations.

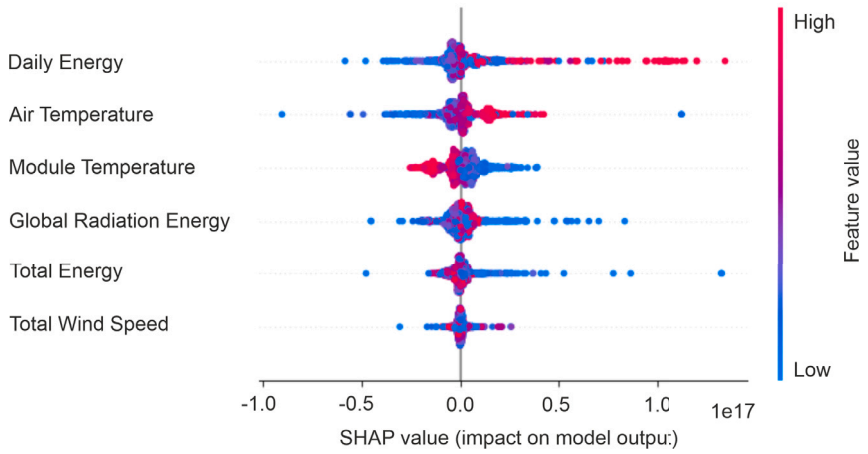

Fig. 6. SHAP summary (global interpretation) plot of the prediction XGBoost model along with its features.

4. Results and discussion

This section presents results and discussions regarding the application of the proposed model. Initially, the configurations used for comparison are presented, followed by the evaluation metrics used in this analysis. After these definitions, the hypertuning results are presented, and additional analyses are discussed in detail.

4.1. Experimental setup

All experiments were conducted on a high-performance workstation running Gentoo Linux. The computational environment includes an Intel(R) Core(TM) i9-14900K, and an NVIDIA GeForce RTX 4090 Graphics Processing Unit (GPU). The system is equipped with 31.03 GiB of Random Access Memory (RAM) and multiple high-capacity storage volumes, ensuring efficient data handling and model training.

4.2. Feature engineering

4.2.1. SHAP analysis

This section employs SHAP [44] to quantify and rank the contribution of individual features to predictions generated by the XGBoost model. Subsequently, evaluate temporal dependencies in the dataset using the autocorrelation and partial autocorrelation functions. These analyses identify redundant or overly correlated features in time-series data, enabling the removal of non-informative predictors while preserving relevant temporal patterns. For [ML, DL]-based models, SHAP provides efficient, exact explanations using `TreeExplainer`. An XGBoost regressor was trained using the standardized hourly dataset, limited to the 5 to 19 h range from December 21, 2021, to December 20, 2022. Fig. 6 presents the SHAP plot obtained by applying `TreeExplainer` to the test set. Each dot represents a SHAP value for an observation; color encodes the original feature value, red: high, and blue: low.

The SHAP results indicate that PV energy exerts the highest contribution to model predictions. Meteorological variables such as air temperature and module temperature also demonstrate non-negligible contributions, with warmer temperatures often positively correlated with predicted energy values. The feature representing global solar radiation shows a moderately symmetric distribution of SHAP values, indicating both positive and negative impacts depending on other features. The distribution of SHAP values across individual features highlights not only the relative magnitude of their influence but also the variability of their impact across different samples [45].

Features such as wind speed and Total energy exhibit a narrow range of SHAP values, suggesting a limited overall influence on the model output within the scope of the dataset. The interpretability provided by SHAP supports both model transparency and feature selection efforts [46]. These results confirm that, for the selected period and data window, generation-related telemetry and temperature readings play a central role in explaining variations in hourly PV power output from solar module B. Similar results were obtained for the other modules.

4.2.2. Autocorrelation analysis

The ACF [47] applied here quantifies the linear relationship between a time series and its lagged values, providing critical insights into temporal dependencies for forecasting models. By analyzing ACF plots, specialists can detect seasonality (peaks at fixed intervals), trends (slowly decaying correlations), and optimal lag structures for AR processes. In ARIMA modeling, the ACF helps identify the order of moving average terms via cutoff points and validates model adequacy by ensuring residuals exhibit no significant autocorrelation (white noise). Additionally, ACF aids in diagnosing non-stationarity, guiding differencing requirements to stabilize variance. Its integration with the PACF further refines parameter selection, ensuring robust time series forecasting.

Fig. 7 presents the ACF and PACF profiles for PV energy, restricted to the 5 am to 7 pm interval from December 21, 2021, to December 20, 2022. The profiles are computed over 42 lags, corresponding to approximately three days of hourly observations, given the 14-hour daily sampling window.

The ACF plot displays a gradual decay, with significant positive correlations observed at lags 1 through 6. This indicates the presence of short-term autocorrelation, suggesting that hourly values are influenced by their immediate predecessors. Additional moderate peaks beyond lag equals 10 may reflect residual seasonal or intra-day effects. The PACF plot exhibits a strong coefficient at lag equals 1, followed by coefficients at higher lags that mostly fall within the 95% confidence interval. This pattern is indicative of a first-order autoregressive structure, where the current value is predominantly explained by its immediately preceding value. The lack of pronounced partial correlations at higher lags suggests that further past observations contribute limited additional predictive energy when conditioned on recent values. Everything considered, the autocorrelation structure of PV energy supports the adoption of low-order autoregressive components in subsequent time series modeling efforts.

4.3. Lags and forecast horizons analysis

Table 3 shows the optimal model performance for all ML and DL models applied with forecast horizons and three lags, 6, 12, and 20. For 1, 2, and 6 h ahead forecasts, the TabPFN achieves optimal performance across all lag window configurations. For 12 and 24 h, two-based

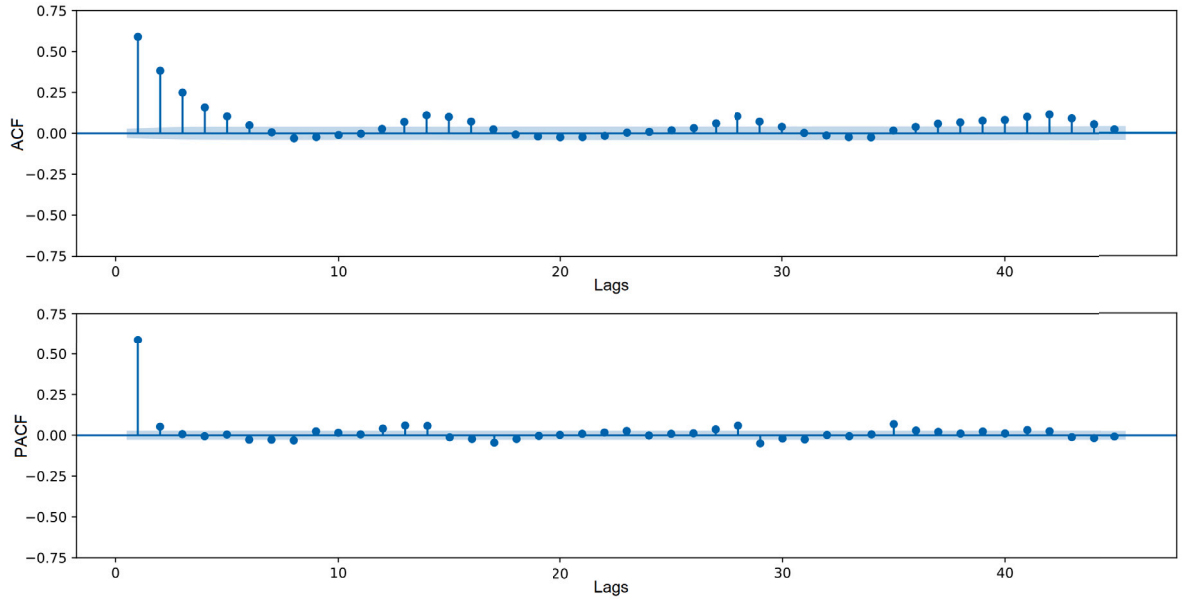

Fig. 7. ACF and PACF plots of the PV energy generation data time series (hourly data, 5 am to 7 pm).

Table 3
Optimal models' performance by forecast horizon and lag window size based on MSE.

Horizon (h)	6 lags	12 lags	20 lags
1	TabPFN (0.066)	TabPFN (0.057)	TabPFN (0.060)
2	TabPFN (0.149)	TabPFN (0.139)	TabPFN (0.136)
6	TabPFN (0.384)	TabPFN (0.417)	TabPFN (0.395)
12	NGBoost (0.678)	RF (0.675)	NGBoost (0.682)
24	NGBoost (0.989)	NGBoost (0.962)	NGBoost (0.995)
48	TabPFN (1.056)	TabPFN (1.148)	NGBoost (1.161)
72	NGBoost (1.151)	NGBoost (1.143)	NGBoost (1.141)
96	TabPFN (1.210)	NGBoost (1.166)	NGBoost (1.169)

ensemble models dominate, RF (0.675) and NGBoost (0.962), respectively. However, for 48, 72, and 96 h, the optimal models are TabPFN, NGBoost, and NGBoost, respectively, with approximately 6, 20, and 12 lags.

The MSE values (see Table 3) increase with the forecast horizon, indicating the expected deterioration in predictive accuracy as the temporal distance increases. However, the optimal lag window size varies across horizons, with no universally optimal setting. Thus, results using the MSE metric indicate that neural network-based models (TabPFN) excel for smaller short-term forecasts and DL-based ensemble learning methods (mainly NGBoost) provide superior performance for larger short-term horizons, and lag 20 will be considered adequate for all forecast horizons.

We also experimented with two representative deep learning sequence-to-sequence architectures under the larger lag configuration (20 lags): a recurrent baseline (LSTM) and a vanilla Transformer encoder–decoder. Table 4 reports their performance across multiple forecast horizons (1–96 h ahead) using MSE, RMSE, and MAE metrics. These results highlight the degradation in accuracy as the prediction horizon increases, with both models showing competitive short-term performance but diverging more substantially at longer horizons.

4.4. Hyperparameter tuning results

To ensure robust and reliable performance evaluation, a time series cross-validation approach with an expanding window strategy was used. Unlike standard k-fold methods that risk data leakage by shuffling observations, this approach preserves the chronological order of the data. For each of the 5 folds, the model was trained on all data up to fold $k-1$, validated on fold k, and tested on fold $k+1$. As the process advanced, the training window expanded while validation and test windows moved forward in time. Fold sizes were dynamically adjusted to align with the sequence of model and prediction length requirements. We conducted a hyperparameter search to identify adequate settings for the proposed FNO-BiLSTM-NAM approach.

Table 4
Performance of LSTM and Transformer models across forecast horizons (20 lags).

Horizon (h)	Model	MSE	RMSE	MAE
1	LSTM	0.0359	0.1895	0.1895
	Transformer	0.0225	0.1500	0.1500
2	LSTM	0.0461	0.2147	0.2133
	Transformer	0.0309	0.1757	0.1741
6	LSTM	0.0432	0.2079	0.1951
	Transformer	0.0334	0.1827	0.1695
12	LSTM	0.0541	0.2325	0.2185
	Transformer	0.0583	0.2414	0.2233
24	LSTM	0.0553	0.2351	0.2134
	Transformer	0.0575	0.2397	0.2121
48	LSTM	0.0859	0.2930	0.2770
	Transformer	0.0980	0.3131	0.2919
72	LSTM	0.0754	0.2746	0.2509
	Transformer	0.0849	0.2914	0.2634
96	LSTM	0.0831	0.2882	0.2588
	Transformer	0.0892	0.2987	0.2683

Table 5 quantifies the relative importance of each hyperparameter through correlation analysis with validation loss. The FNO dimension has the highest importance score of 0.541, suggesting that the spectral representation capacity affects model performance. The hidden dimension ranks second with a score of 0.308, followed by teacher forcing decay (0.151), dropout (0.148), number of layers (0.069), and initial teacher forcing ratio (0.020).

The analysis reveals that the FNO dimension has a positive correlation with validation loss, indicating that higher dimensions tend to increase error, potentially due to overfitting. Conversely, the hidden dimension shows a negative correlation, where larger values generally improve performance by enhancing the representational capacity of the

Table 5
Relative importance of hyperparameters based on correlation with validation loss.

Hyperparameter	Correlation type	Importance (|correlation|)
rnn_dim	Negative	**0.541**
dropout	Negative	0.148
num_layers	Negative	0.069
hidden_dim	Positive	0.308
tf_decay	Positive	0.151
initial_tf_ratio	Positive	**0.020**

Negative correlation higher is better, and Positive correlation lower is better.

Table 6
Hyperparameters search space and optimized values to FNO-BiLSTM-NAM.

Hyperparameter	Domain	Optimized value
initial_tf_ratio	$\mathcal{U}(0.3, 0.9)$[a]	0.394
tf_decay	$\mathcal{U}(0.95, 0.99)$[a]	0.952
fno_dim	{32, 64, 128}	128
num_layers	{2, 3, 4}	4
hidden_dim	{64, 128, 256}	64
dropout	$\mathcal{U}(0.1, 0.5)$[a]	0.497

[a] Where $\mathcal{U}(a, b)$ denotes a uniform distribution over the interval $[a, b]$.

model. The low importance of the number of layers and initial teacher forcing ratio suggests that the model architecture is less sensitive to these parameters within the ranges explored. Based on this analysis, the search space for the hyperparameters and the configuration that minimizes validation loss for the final proposed model was selected, as shown in Table 6.

4.5. Fourier neural operator analysis

The spectral characteristics of the FNO component reveal several patterns in how the model processes solar energy forecasting data. Fig. 8 presents the overall spectral transformation performed by the FNO. The top row shows a time-domain representation of input features and FNO output channels. The middle row displays FFT magnitude heatmaps for inputs (left) and outputs (right). The bottom row shows FFT magnitude comparison (left) and normalized energy spectrum comparison (right), with complex weight magnitudes and phase

The time-domain comparison (Fig. 8, top row) demonstrates that the FNO transforms high-frequency fluctuations in the input features into smoother output signals. The Total Energy and Global Radiation features exhibit the most pronounced fluctuations, which are attenuated in the output channels. This filtering effect is quantified in the FFT magnitude heatmaps, where high-frequency components present in the input features (middle row, left) are reduced in the output channels (middle row, right).

The normalized energy spectrum (Fig. 8, bottom row, right) confirms this low-pass filtering behavior, with the output power concentrated in the lowest frequency modes (0–2). The model allocates 60% of the spectral power to the DC component (mode 0) and approximately 15% to the first harmonic (mode 1). This distribution indicates that the FNO primarily models long-term trends while filtering out high-frequency fluctuations, which may represent noise or short-term variations irrelevant to the prediction task.

The complex weight magnitudes (Fig. 8, middle row, left) show stronger weights for low-frequency modes across most input–output channel pairs. The polar phase visualization (Fig. 8, middle row, right) reveals phase alignment patterns in the learned spectral filters, with several input-mode pairs exhibiting similar phase angles, suggesting coherent transformation of certain frequency components.

Figs. 9 and 10 detail the feature-specific Fourier characteristics. The feature-specific analysis reveals different spectral profiles across input variables. The Global Radiation feature shows the highest magnitude in frequency mode 0, indicating a strong DC component reflective of daily solar patterns.

Other meteorological features, such as air temperature and module temperature, exhibit more distributed frequency content, but still predominantly in the lower modes. The feature attention weights (Figs. 9 and 10, leftmost column) indicate that the model assigns comparable attention values to all features, with slightly higher attention to Global Radiation, which aligns with its physical relevance to solar power generation.

The weighted frequency response (rightmost column) shows consistent patterns across features, with higher activation in approximately 10 to 15% of the output channels at low-frequency modes. The frequency domain analysis (Figs. 9 and 10, third column) demonstrates that most features exhibit a rapid decay in magnitude after the first few modes. Weather-related features (global radiation, air temperature, wind speed) show more energy in higher-frequency components compared to energy measurement features, reflecting their higher temporal variability.

Together, these results indicate that the FNO component conducts spectrally aware feature transformation, with a preferential emphasis on low-frequency components. This behavior suggests that the model has learned to distinguish between predictive temporal patterns and high-frequency noise, focusing on the most informative spectral modes for solar power forecasting.

4.6. Neural additive model analysis

As previously stated, the model architecture incorporates an NAM block running in parallel with the FNO-BiLSTM components. In the final model configuration, the FNO-BiLSTM component was assigned a relative weight of 0.60, while the NAM block contributed the remaining 0.40 to the overall prediction. This weighting reflects the complementary nature of these components, where the FNO-BiLSTM captures complex temporal dependencies while the NAM block models individual feature contributions. In the FNO-BiLSTM framework, the BiLSTM is used to process sequential data in both forward and backward directions, enabling it to capture contextual dependencies from past and future states simultaneously.

Fig. 11 shows the feature importance analysis within the NAM block. Global radiation exhibits the highest importance at 0.184, followed by the target variable's historical values, PV energy at 0.167. These two features collectively account for 35.1% of the NAM block's predictive power. Air temperature ranks third with an importance of 0.096, followed by energy data from alternative sensors at 0.085. Weather-related variables demonstrate moderate importance: cloud cover, 0.073, precipitation, 0.064, and wind speed, 0.059. Module temperature and total energy from secondary loggers show lower importance values at 0.058 and 0.034, respectively.

This importance distribution aligns with physical energy generation principles, where solar radiation and historical production patterns serve as primary predictors, followed by temperature effects and meteorological conditions. The NAM block effectively captures these individual feature contributions, complementing the temporal pattern modeling provided by the FNO-BiLSTM component. The parallel architecture with a weighted combination provides two advantages. It leverages both temporal dependencies and feature-specific effects, and offers interpretability through the proposed framework's explicit feature importance values while maintaining the predictive power of deep learning components.

4.7. Ablation study

Table 7 presents the results of our ablation study, where each model component was systematically removed to assess its individual contribution to forecasting performance. The study evaluated six key architectural components across three performance metrics: MAE,

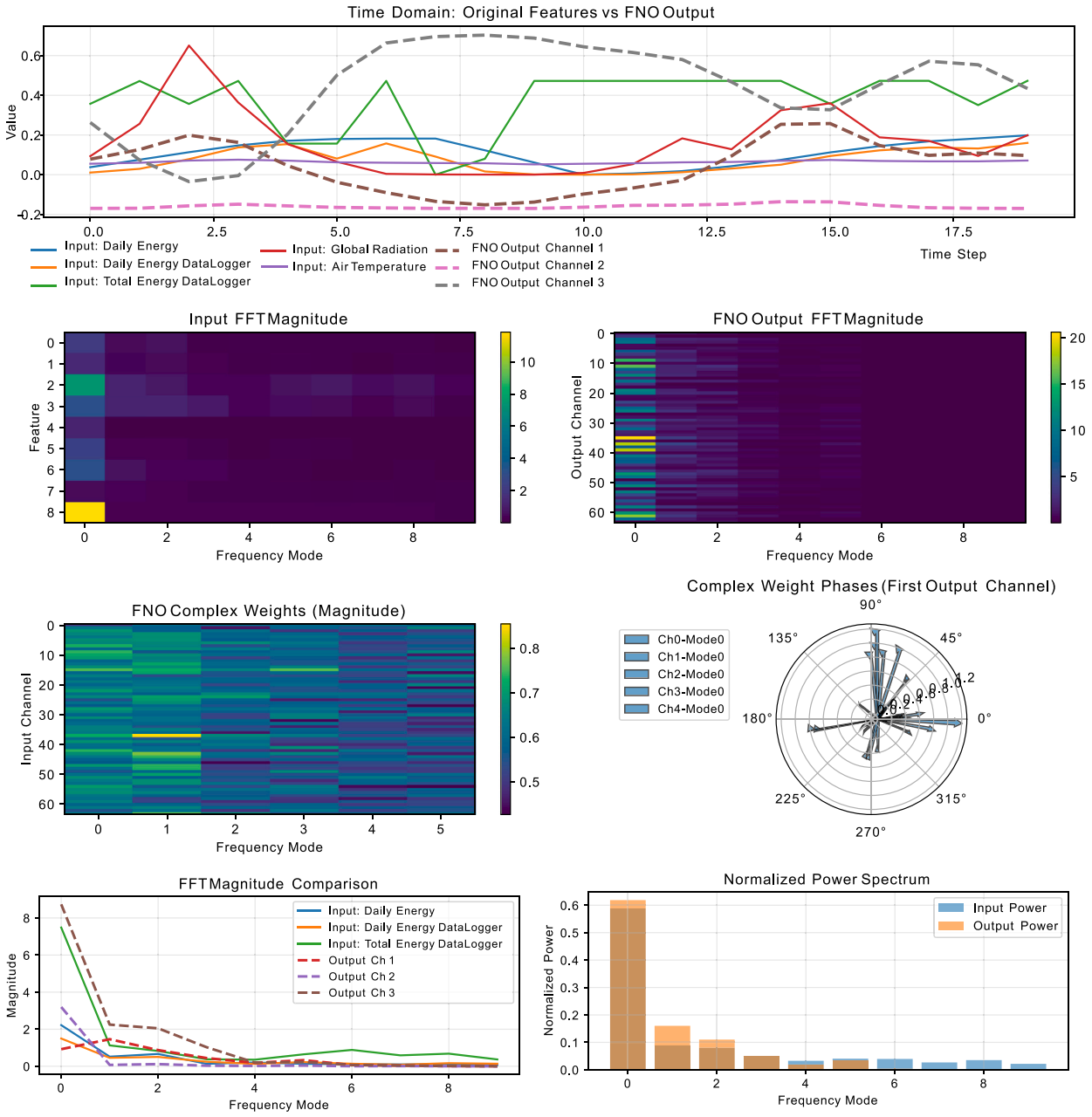

Fig. 8. Spectral transformation analysis of the FNO component.

RMSE, and MAPE. The FNO demonstrates the largest impact on model performance. Removing this component results in a 67% increase in MAE (from 0.0865 to 0.1448) and a 49% increase in RMSE (from 0.1320 to 0.1971), indicating its critical role in capturing the spectral characteristics of solar energy generation patterns. The FNO's effectiveness stems from its ability to learn solution operators in the frequency domain, which aligns with the periodic nature of solar irradiance and energy production cycles.

Feature attention mechanisms show consistent impact across all metrics, with removal increasing MAE by 11% and RMSE by 8.6%, while MAPE remains stable. This suggests that feature attention improves the model's ability to focus on relevant input variables, particularly during periods of high variability in meteorological conditions. Similarly, NAM components exhibit distinct effects on percentage-based errors. While the absolute metrics show moderate degradation (15% and 12% increases in MAE and RMSE, respectively), MAPE increases from 90.44% to 114.92%, indicating that NAM components are effective at maintaining relative accuracy across different scales of energy generation.

In contrast, teacher forcing removal produces limited performance degradation, with MAE increasing by only 5.4% and RMSE by 3.1%. The modest impact suggests that the model has developed internal representations to maintain accuracy without explicit ground-truth guidance during training sequences. Component weighting mechanisms show variable impact across metrics, with moderate increases in MAE and RMSE (8.6% and 6.1% respectively) but a substantial increase in MAPE to 128.46%, indicating that adaptive component weighting maintains consistent relative accuracy across different operational conditions.

The comparison between bidirectional and unidirectional processing reveals a performance trade-off. The unidirectional configuration achieves better MAPE performance (79.86% vs 90.44%), representing a 12% improvement in relative accuracy, but at the cost of degraded absolute error metrics, with MAE increasing by 2.8% and RMSE by 5.0%.

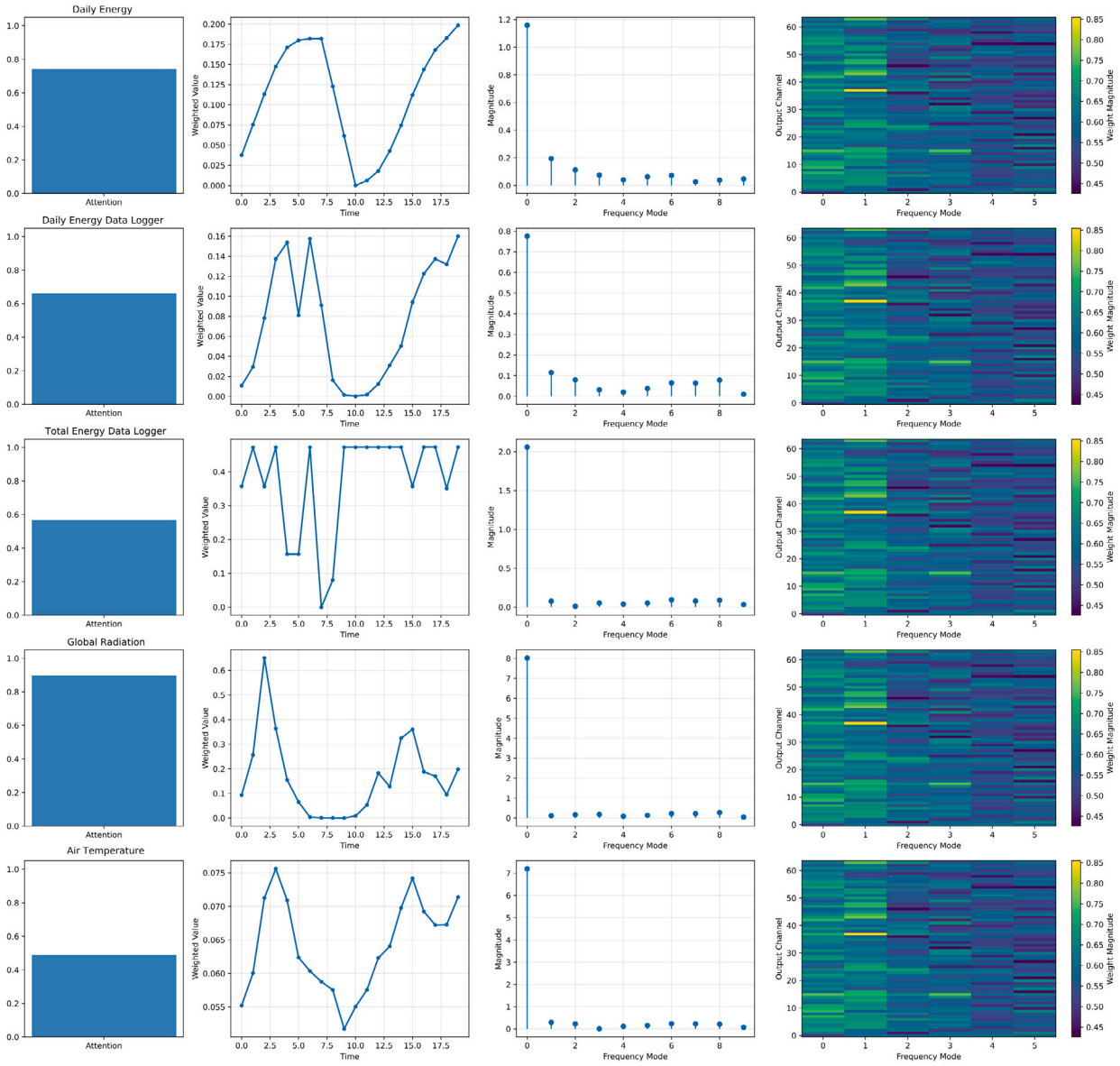

Fig. 9. Feature-specific Fourier analysis A.

This trade-off suggests that bidirectional processing may introduce regularization effects that improve absolute error metrics but reduce relative accuracy, while the unidirectional approach appears more effective at preserving proportional relationships in energy generation predictions.

The full model configuration achieves optimal performance in absolute error metrics (MAE and RMSE), demonstrating positive synergistic effects between components. However, the fact that the unidirectional configuration outperforms the full model in MAPE suggests that component interactions can be complex and metric-dependent. The results indicate that while individual components contribute positively to model performance, their interactions may introduce trade-offs between different types of accuracy, highlighting the importance of considering multiple evaluation metrics when designing neural architectures for solar forecasting applications.

These findings provide guidance for solar forecasting model architecture decisions. The FNO component emerges as indispensable, likely due to its alignment with the periodic nature of solar energy systems. Feature attention and NAM components provide consistent improvements and should be retained in production systems, while the choice between bidirectional and unidirectional processing should be guided by specific application requirements, with unidirectional processing preferred when relative accuracy is paramount.

4.8. Analysis of FNO-BiLSTM-NAM performance

This section presents a quantitative evaluation of the proposed FNO-BiLSTM-NAM against established [ML,DL]-based forecasting models, analyzing performance across multiple temporal horizons using standard error metrics.

Table 8 presents the performance results of the FNO-BiLSTM-NAM model for different forecast horizons (1 to 96 h), with error metrics MSE, RMSE, and MAE indicating that the model performs best for short-term forecasts, especially at 6 h, where the errors reach their lowest values, MSE = 0.0092 and MAE = 0.0712. From 12 h onwards, the errors increase progressively until 72 h, reflecting the greater uncertainty inherent in long-term forecasting. However, between 72 h and 96 h, a slight stabilization in errors is observed, suggesting that the model maintains relatively consistent performance even over extended horizons. These results indicate that the model is effective in the short and medium terms and is robust to longer forecasts, although with a loss of accuracy.

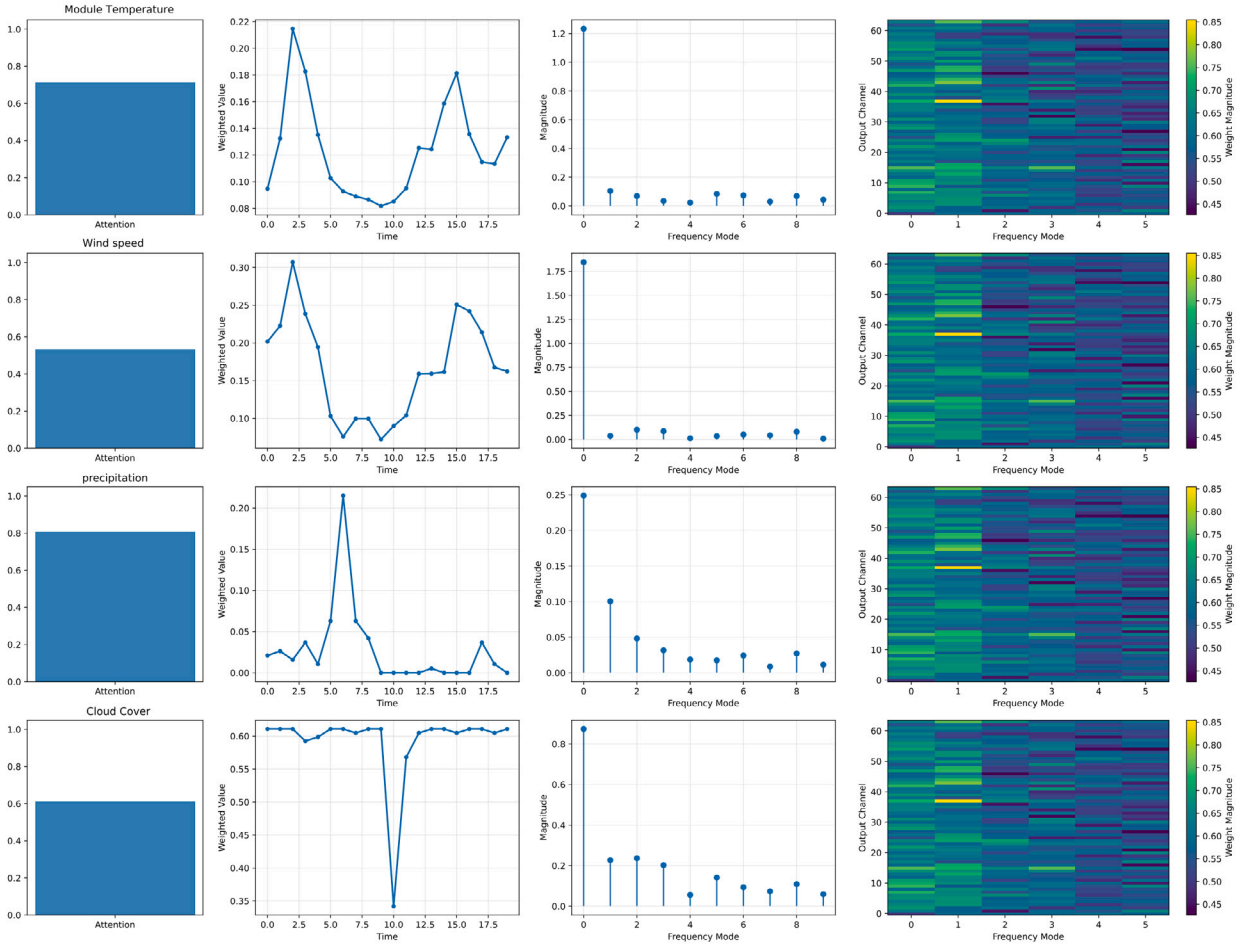

Fig. 10. Feature-specific Fourier analysis B.

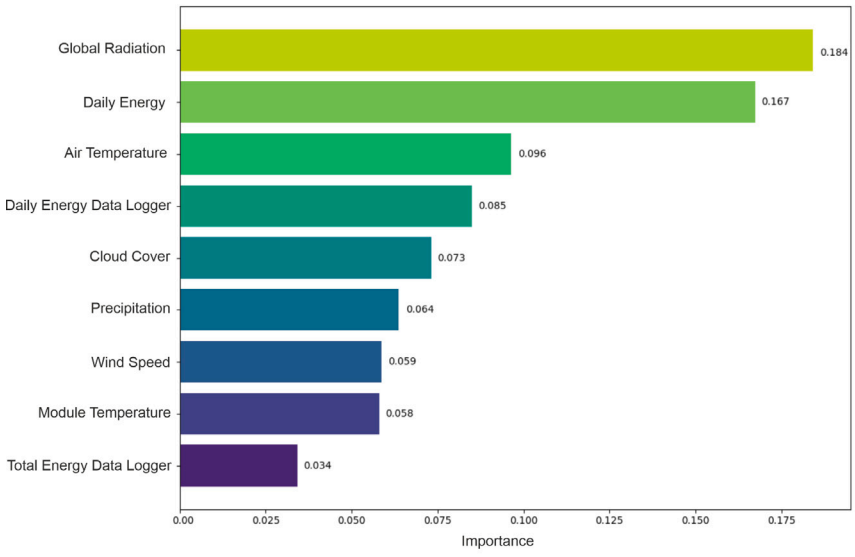

Fig. 11. Feature importance analysis over the dataset in the NAM block.

Table 7
Ablation study results: Impact of model components on solar forecasting performance.

Feature attention	FNO	NAM	Teacher forcing	Component weighting	Bidirectional	MAE↓	RMSE↓	MAPE↓ (%)
✓	✓	✓	✓	✓	✓	**0.0865**	**0.1320**	90.44
✗	✓	✓	✓	✓	✓	0.0960	0.1433	90.39
✓	✗	✓	✓	✓	✓	0.1448	0.1971	109.10
✓	✓	✗	✓	✓	✓	0.0997	0.1474	114.92
✓	✓	✓	✗	✓	✓	0.0912	0.1361	93.77
✓	✓	✓	✓	✗	✓	0.0939	0.1401	128.46
✓	✓	✓	✓	✓	✗	0.0889	0.1386	**79.86**

Best performance in each metric highlighted in bold.

Table 8
Comparison of performance metrics (mean ± Std) and forecast horizons for the test set of solar PV forecasting obtained by FNO-BiLSTM-NAM.

Horizon (h)	MSE	RMSE	MAE
1	0.0129 ± 0.0170	0.0856 ± 0.0742	0.0856 ± 0.0742
2	0.0139 ± 0.0191	0.0911 ± 0.0750	0.0898 ± 0.0760
6	0.0092 ± 0.0124	0.0770 ± 0.0568	0.0712 ± 0.0547
12	0.0375 ± 0.0022	0.1935 ± 0.0057	0.1461 ± 0.0138
24	0.0436 ± 0.0122	0.2068 ± 0.0292	0.1649 ± 0.0180
48	0.0511 ± 0.0062	0.2257 ± 0.0141	0.1822 ± 0.0108
72	0.0685 ± 0.0059	0.2616 ± 0.0115	0.1973 ± 0.0101
96	0.0665 ± 0.0054	0.2576 ± 0.0106	0.1969 ± 0.0093

The RMSE and MAE metrics increase with the forecast horizon until 72 h, followed by a slight decrease at 96 h. This pattern aligns with expected uncertainty growth in medium-term forecasting while suggesting potential cyclical or seasonal effects that facilitate improved long-term predictions. The standard deviations of these metrics generally decrease as the forecast horizon extends, indicating more consistent performance for longer-term predictions over different time periods.

Table 9 provides a comprehensive comparison of the FNO-BiLSTM-NAM to other models across different prediction horizons ranging from 1 to 96 h. The analysis focuses primarily on comparing error reduction ratio, ρ, e.g., improvement, achieved by the proposed FNO-BiLSTM-NAM compared to the best [ML,DL]-based model at horizon h, which can be expressed as:

$$\rho = 1 - \frac{\mathrm{MSE}_{\mathrm{neural}}(h)}{\mathrm{MSE}_{[\mathrm{ML,DL}]}(h)}. \tag{32}$$

The results demonstrate that the proposed FNO-BiLSTM-NAM model consistently outperforms both LSTM and Transformer baselines across all forecast horizons. The relative improvement (ρ) ranges from 9.1% at 72 h to 72.5% at 6 h, with the largest gains observed in the short-term horizons (1–6 h). At these intervals, the model achieves MSE values of 0.0129, 0.0139, and 0.0092 for 1, 2, and 6 h ahead, corresponding to improvements of 42.7%, 55.0%, and 72.5% over the best competing transformer models.

At the 12-hour horizon, the model records an MSE of 0.0375 ± 0.0022, achieving a 30.7% improvement compared to the LSTM. In the medium-term horizons of 24 and 48 h, the FNO-BiLSTM-NAM continues to reduce errors relative to the best competing models, achieving MSE values of 0.0436 and 0.0511, which represent improvements of 21.2% and 40.5% compared to the LSTM. For long-term horizons (72–96 h), the performance gap narrows but remains consistent. The proposed model achieves MSE values of 0.0685 and 0.0665, which translate into improvements of 9.1% and 20.0% relative to the LSTM. These results indicate that while forecasting accuracy naturally decreases with longer horizons, the proposed FNO-BiLSTM-NAM maintains a systematic advantage over established sequence models across all time scales.

To illustrate the practical application of FNO-BiLSTM-NAM in operational settings, was implemented an autoregressive forecasting approach across one week (2022-W47). The next figure show the results where each 8-hour prediction window is recursively fed back into the model to generate subsequent forecasts. The vertical dashed lines in these figures show the boundaries between consecutive autoregressive cycles, highlighting ability of the model to maintain prediction stability across multiple recursive steps. Each cycle incorporates the previous predictions as inputs to the subsequent forecast window, simulating real-world operational conditions where ground truth data is unavailable for future periods.

Table 9
Performance comparison of the proposed FNO-BiLSTM-NAM model against the best competing model (LSTM or Transformer) for the test set of solar PV energy forecasting. Values correspond to MSE (mean ± Std). The ρ column reports the relative improvement of FNO-BiLSTM-NAM with respect to the best competing model at each horizon.

Horizon (h)	FNO-BiLSTM-NAM	Best competing model	ρ (%)
1	0.0129 ± 0.0170	0.0225 (Transformer)	42.7
2	0.0139 ± 0.0191	0.0309 (Transformer)	55.0
6	0.0092 ± 0.0124	0.0334 (Transformer)	72.5
12	0.0375 ± 0.0022	0.0541 (LSTM)	30.7
24	0.0436 ± 0.0122	0.0553 (LSTM)	21.2
48	0.0511 ± 0.0062	0.0859 (LSTM)	40.5
72	0.0685 ± 0.0059	0.0754 (LSTM)	9.1
96	0.0665 ± 0.0054	0.0831 (LSTM)	20.0

As previously discussed, the forecast utility was enhanced by generating multiple scenarios based on different exogenous variable assumptions, a low scenario (q25) representing conservative estimates, an average scenario (q50) reflecting the expected central tendency, and a high scenario (q75) capturing optimistic projections. Additionally, the confidence intervals (q25–q75) to quantify forecast uncertainty were provided. The mean of scenarios (dashed black line) often approximates the ground truth (solid black line) more accurately than any individual scenario, demonstrating the value of ensemble approaches in operational forecasting.

Fig. 12 illustrates the multi-scenario forecast for the PV energy prediction task during the week of 2022-W47 (from November 21 to November 27, 2022), comparing the average, high, and low scenarios (q50, q75, and q25) against the ground truth values. The neural network generates recursive 8-hour predictions (separated by vertical dashed lines) with three scenarios based on varying exogenous inputs: average (blue), high (green), and low (red). The shaded blue area represents the q25–q75 quantiles confidence interval. The black solid line shows ground truth values, while the black dashed line indicates the mean of all scenarios.

The shaded confidence interval between the 25th and 75th percentiles effectively captures the uncertainty in the forecasts. Throughout the week, the predicted average scenario closely follows the actual values, especially during peak hours, demonstrating strong alignment in trend and magnitude. While some deviations occur – particularly underestimations and overestimations during valleys and peaks – these are generally bounded within the confidence interval, highlighting robustness. Additionally, the mean of scenarios stays within a narrow band around the ground truth, further validating the consistency and reliability of the forecasting framework.

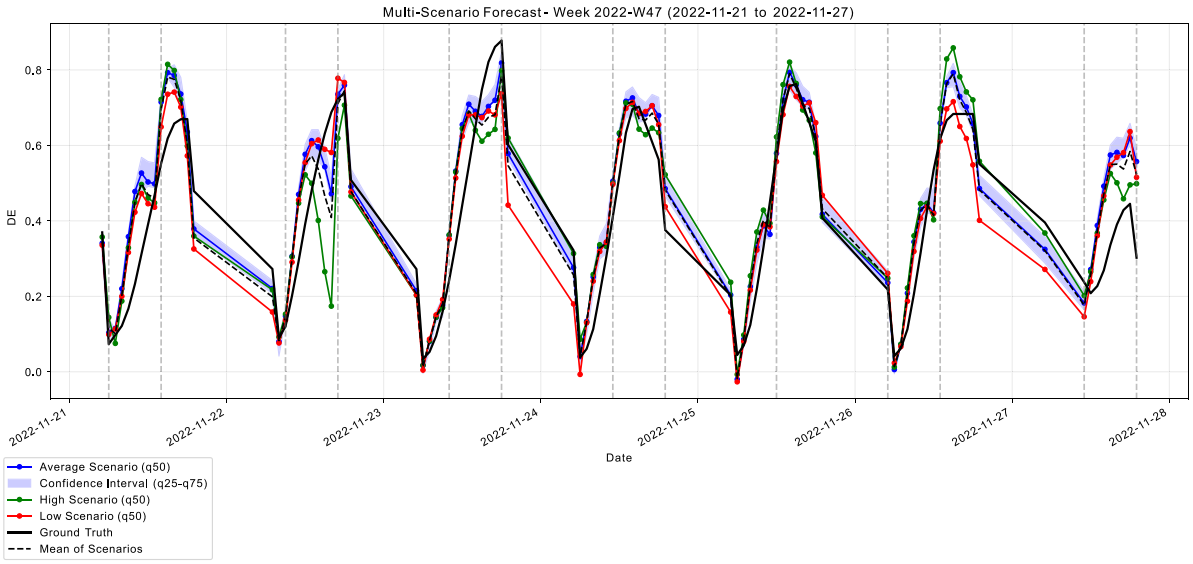

Fig. 12. PV energy multi-scenario forecast for week 2022-W47 (2022-11-21 to 2022-11-27).

As evidenced in the figure, the proposed FNO-BiLSTM-NAM model successfully captures both daily periodicity and inter-day variations in energy production patterns. The autoregressive implementation maintains prediction accuracy even at extended horizons, with no significant error accumulation or drift observed across the 7-day forecast periods. This stability contrasts with typical autoregressive models, where error propagation often degrades long-term forecast quality. The ability of the model to track sudden changes in production patterns (such as the sharp peaks observed on 2022-11-24) while maintaining reasonable confidence intervals demonstrates its robustness to anomalous conditions.

4.9. Comparison to other research

In comparison to our model, Sabadus et al. [2] achieved an MAPE of 24.62% and a normalized RMSE of 10.54% for PV power forecasting. This result highlighted the promising performance of state-of-the-art models. They had different results depending on the climate (tropical, arid, and temperate), proving that the performance of the model is closely related to the data being analyzed. With this in mind, direct comparison between models using different data is not an adequate analysis. However, we have proven that it is possible to achieve better MAPE results using the NAM (24.48%), which is an approach that could be explored in other studies.

Brester et al. [3] had an MAE of 197.67 for the baseline, 262.27 using the alternative, and 138.94 considering optimistic, proving that optimization is indeed useful. Our optimized model had an MAE of 0.0856 for one step ahead, confirming the improvement potential of the optimized model. Considering a hybrid model, Thaker and Höller [4] had a normalized MAE of 6.65%, proving that the hybrid approach is a promising solution, as carried out in our model. Through our ablation study, we prove that the use of several combined techniques surpasses their individual use, agreeing with what has been proposed by other authors, such as the work of Kushwaha and Pindoriya [5] or the model proposed by Jacques Molu et al. [14].

4.10. Limitations

A primary drawback of the proposed FNO-BiLSTM-NAM is the requirement for access to high-quality historical and meteorological data. In scenarios where data are incomplete, noisy, or unavailable, forecasting accuracy can deteriorate, reducing the model's reliability. This dependency on extensive data collection infrastructures may pose challenges for small-scale solar installations or regions with limited monitoring resources.

Another concern lies in the risk of overfitting of the proposed model. The high number of trainable parameters, combined with its multi-component structure, increases sensitivity to dataset-specific patterns, which can impair performance when applied to different PV plants or climatic conditions. While the incorporation of a NAM improves interpretability to some extent, the overall system remains intricate, which may hinder full transparency and make it difficult for stakeholders to fully understand the decision-making process.

5. Conclusion

This study proposed a novel hybrid forecasting framework, FNO-BiLSTM-NAM, to address limitations in current PV energy forecasting methods by integrating three specialized neural network components: the Fourier Neural Operator, Bidirectional Long Short-Term Memory, and Neural Additive Model. The model captures long-range temporal patterns, local sequential dependencies, and interpretable feature contributions, enabling accurate and robust multi-horizon probabilistic forecasting.

Comprehensive experiments were conducted on a real PV dataset collected from a 5 MW solar power plant in southern Brazil. The proposed framework achieved superior performance across short and medium term horizons, with the lowest errors recorded at the 6-hour horizon (MAE = 0.0712), confirming its high predictive accuracy. Furthermore, performance was evaluated for extended forecasts up to 96 h, revealing a trade-off between horizon length and accuracy, yet maintaining stable error behavior. The framework also introduces a multi-scenario forecasting mechanism (low, average, and high projections), which supports more informed decision-making in solar energy operations under uncertainty.

The proposed framework improved interpretability by integrating an NAM, which quantifies the contribution of individual features such as radiation, temperature, and wind speed to the forecasts. This feature-wise attribution, supported by SHAP analysis, allows operators to clearly understand the influence of meteorological and operational variables. Greater interpretability directly strengthens predictive accuracy by guiding the model to emphasize physically meaningful variables and avoid spurious correlations. The ablation study confirms this, showing that removing the NAM significantly increases error. By combining interpretable feature effects with robust temporal modeling, the framework achieves both transparency and state-of-the-art accuracy across forecasting horizons.

Benchmarking against state-of-the-art ML and DL models, e.g., TabPFN and NGBoost, confirmed the competitive advantage of the proposed architecture. The combination of spectral learning, sequence modeling, and additive explainability makes FNO-BiLSTM-NAM a promising solution for real-time solar forecasting systems. While the model demonstrates robust performance, future work should assess its adaptability to other renewable energy domains, e.g., wind or hybrid systems, explore alternative probabilistic architectures, and evaluate generalization in geographically and climatologically diverse environments. For future work, the proposed model can be evaluated over long periods, focusing on analyzing the influence of seasonality on measurements throughout the year.

CRediT authorship contribution statement

Laio Oriel Seman: Writing – original draft, Software, Investigation, Formal analysis, Conceptualization. **Stefano Frizzo Stefenon:** Writing – review & editing, Methodology. **Kin-Choong Yow:** Writing – review & editing, Supervision. **Leandro dos Santos Coelho:** Writing – review & editing, Supervision. **Viviana Cocco Mariani:** Writing – review & editing, Supervision.

Declaration of competing interest

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

Data availability

Data will be made available on request.

References

[1] U.R.J. Eiva, T.A. Chowdury, S.S. Islam, A. Ullah, J.N. Tuli, M.T. Islam, Comprehensive analysis of fixed tilt and dual-axis tracking photovoltaic systems for enhanced grid integration and energy efficiency, Renew. Energy 256 (2025) 123865, http://dx.doi.org/10.1016/j.renene.2025.123865.

[2] A. Sabadus, R. Blaga, S.M. Hategan, D. Calinoiu, E. Paulescu, O. Mares, R. Boata, N. Stefu, M. Paulescu, V. Badescu, A cross-sectional survey of deterministic PV power forecasting: Progress and limitations in current approaches, Renew. Energy 226 (2024) 120385, http://dx.doi.org/10.1016/j.renene.2024.120385.

[3] C. Brester, V. Kallio-Myers, A.V. Lindfors, M. Kolehmainen, H. Niska, Evaluating neural network models in site-specific solar PV forecasting using numerical weather prediction data and weather observations, Renew. Energy 207 (2023) 266–274, http://dx.doi.org/10.1016/j.renene.2023.02.130.

[4] J. Thaker, R. Höller, Hybrid model for intra-day probabilistic PV power forecast, Renew. Energy 232 (2024) 121057, http://dx.doi.org/10.1016/j.renene.2024.121057.

[5] V. Kushwaha, N.M. Pindoriya, A SARIMA-RVFL hybrid model assisted by wavelet decomposition for very short-term solar PV power generation forecast, Renew. Energy 140 (2019) 124–139, http://dx.doi.org/10.1016/j.renene.2019.03.020.

[6] I. Tavares, R. Manfredini, J. Almeida, J. Soares, S. Ramos, Z. Foroozandeh, Z. Vale, Comparison of PV power generation forecasting in a residential building using ANN and DNN, IFAC-PapersOnLine 55 (9) (2022) 291–296, http://dx.doi.org/10.1016/j.ifacol.2022.07.051, 11th IFAC Symposium on Control of Power and Energy Systems CPES 2022.

[7] W. Khan, S. Walker, W. Zeiler, Improved solar photovoltaic energy generation forecast using deep learning-based ensemble stacking approach, Energy 240 (2022) 122812, http://dx.doi.org/10.1016/j.energy.2021.122812.

[8] H. Ma, C. Zhang, T. Peng, M.S. Nazir, Y. Li, An integrated framework of gated recurrent unit based on improved sine cosine algorithm for photovoltaic power forecasting, Energy 256 (2022) 124650, http://dx.doi.org/10.1016/j.energy.2022.124650.

[9] D.V. Pombo, H.W. Bindner, S.V. Spataru, P.E. Sørensen, P. Bacher, Increasing the accuracy of hourly multi-output solar power forecast with physics-informed machine learning, Sensors 22 (3) (2022) http://dx.doi.org/10.3390/s22030749.

[10] G. Etxegarai, A. López, N. Aginako, F. Rodríguez, An analysis of different deep learning neural networks for intra-hour solar irradiation forecasting to compute solar photovoltaic generators' energy production, Energy Sustain. Dev. 68 (2022) 1–17, http://dx.doi.org/10.1016/j.esd.2022.02.002.

[11] F. Rodríguez, I. Azcárate, J. Vadillo, 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. 137 (2022) 107777, http://dx.doi.org/10.1016/j.ijepes.2021.107777.

[12] J. Wang, Y. Zhou, Z. Li, Hour-ahead photovoltaic generation forecasting method based on machine learning and multi objective optimization algorithm, Appl. Energy 312 (2022) 118725, http://dx.doi.org/10.1016/j.apenergy.2022.118725.

[13] Y. Cao, G. Liu, D. Luo, D.P. Bavirisetti, G. Xiao, Multi-timescale photovoltaic power forecasting using an improved stacking ensemble algorithm based LSTM-Informer model, Energy 283 (2023) 128669, http://dx.doi.org/10.1016/j.energy.2023.128669.

[14] R.J. Jacques Molu, B. Tripathi, W.F. Mbasso, S.R. Dzonde Naoussi, M. Bajaj, P. Wira, V. Blazek, L. Prokop, S. Misak, Advancing short-term solar irradiance forecasting accuracy through a hybrid deep learning approach with Bayesian optimization, Results Eng. 23 (2024) 102461, http://dx.doi.org/10.1016/j.rineng.2024.102461.

[15] N.H.A. Rahman, M.Z. Hussin, S.I. Sulaiman, M.A. Hairuddin, E.H.M. Saat, Univariate and multivariate short-term solar power forecasting of 25MWac pasir gudang utility-scale photovoltaic system using LSTM approach, Energy Rep. 9 (2023) 387–393, http://dx.doi.org/10.1016/j.egyr.2023.09.018.

[16] T. Limouni, R. Yaagoubi, K. Bouziane, K. Guissi, E.H. Baali, Accurate one step and multistep forecasting of very short-term PV power using LSTM-TCN model, Renew. Energy 205 (2023) 1010–1024, http://dx.doi.org/10.1016/j.renene.2023.01.118.

[17] M. Gupta, A. Arya, U. Varshney, J. Mittal, A. Tomar, A review of PV power forecasting using machine learning techniques, Prog. Eng. Sci. 2 (1) (2025) 100058, http://dx.doi.org/10.1016/j.pes.2025.100058.

[18] Y. Cao, P. Yong, J. Yu, Z. Yang, Stacking algorithm based framework with strong generalization performance for ultra-short-term photovoltaic power forecasting, Energy 322 (2025) 135599, http://dx.doi.org/10.1016/j.energy.2025.135599.

[19] U. Singh, S. Singh, S. Gupta, M.A. Alotaibi, H. Malik, Forecasting rooftop photovoltaic solar power using machine learning techniques, Energy Rep. 13 (2025) 3616–3630, http://dx.doi.org/10.1016/j.egyr.2025.03.005.

[20] Y. Sun, V. Venugopal, A.R. Brandt, Short-term solar power forecast with deep learning: Exploring optimal input and output configuration, Sol. Energy 188 (2019) 730–741, http://dx.doi.org/10.1016/j.solener.2019.06.041.

[21] D. Korkmaz, SolarNet: A hybrid reliable model based on convolutional neural network and variational mode decomposition for hourly photovoltaic power forecasting, Appl. Energy 300 (2021) 117410, http://dx.doi.org/10.1016/j.apenergy.2021.117410.

[22] M. Gao, J. Li, F. Hong, D. Long, Day-ahead power forecasting in a large-scale photovoltaic plant based on weather classification using LSTM, Energy 187 (2019) 115838, http://dx.doi.org/10.1016/j.energy.2019.07.168.

[23] Y. Jung, J. Jung, B. Kim, S. Han, Long short-term memory recurrent neural network for modeling temporal patterns in long-term power forecasting for solar PV facilities: Case study of South Korea, J. Clean. Prod. 250 (2020) 119476, http://dx.doi.org/10.1016/j.jclepro.2019.119476.

[24] H. Henderi, T. Wahyuningsih, E. Rahwanto, Comparison of min-max normalization and Z-score normalization in the K-nearest neighbor (kNN) algorithm to test the accuracy of types of breast cancer, Int. J. Inform. Inf. Syst. 4 (1) (2021) 13–20, http://dx.doi.org/10.47738/ijiis.v4i1.73.

[25] A. Nguyen, K. Pham, D. Ngo, T. Ngo, L. Pham, An analysis of state-of-the-art activation functions for supervised deep neural network, ICSSE, in: 2021 International Conference on System Science and Engineering, vol. 1, IEEE, Ho Chi Minh City, Vietnam, 2021, pp. 215–220, http://dx.doi.org/10.1109/ICSSE52999.2021.9538437.

[26] Z. Li, D.Z. Huang, B. Liu, A. Anandkumar, Fourier neural operator with learned deformations for pdes on general geometries, J. Mach. Learn. Res. 24 (388) (2023) 1–26.

[27] D. Yu, X. Zhan, L.j. Yang, Y. Jia, Theoretical description of logical stochastic resonance and its enhancement: Fast Fourier transform filtering method, Phys. Rev. E 108 (1) (2023) 014205, http://dx.doi.org/10.1103/PhysRevE.108.014205.

[28] K. Xiong, G. Zhao, G. Shi, Radar high-speed target coherent detection method based on modified radon inverse Fourier transform, IEEE Trans. Aerosp. Electron. Syst. 59 (2) (2023) 950–962, http://dx.doi.org/10.1109/TAES.2022.3193090.

[29] L. Mishra, B. Dinesh, P. Kavyassree, N. Mishra, A Google trend enhanced deep learning model for the prediction of renewable energy asset price, Knowl.-Based Syst. 308 (2025) 112733, http://dx.doi.org/10.1016/j.knosys.2024.112733.

[30] N.E. Michael, R.C. Bansal, A.A.A. Ismail, A. Elnady, S. Hasan, A cohesive structure of bi-directional long-short-term memory (BiLSTM)-GRU for predicting hourly solar radiation, Renew. Energy 222 (2024) 119943, http://dx.doi.org/10.1016/j.renene.2024.119943.

[31] M. Arefi, A.H. Khammar, Nonlinear prediction of fuzzy regression model based on quantile loss function., Soft Comput.-A Fusion Found. Methodol. Appl. 28 (6) (2024) http://dx.doi.org/10.1007/s00500-023-09190-w.

[32] K.G. Olivares, C. Challu, G. Marcjasz, R. Weron, A. Dubrawski, Neural basis expansion analysis with exogenous variables: Forecasting electricity prices with NBEATSx, Int. J. Forecast. 39 (2) (2023) 884–900, http://dx.doi.org/10.1016/j.ijforecast.2022.03.001.

[33] J. Kaur, K.S. Parmar, S. Singh, Autoregressive models in environmental forecasting time series: A theoretical and application review, Environ. Sci. Pollut. Res. 30 (8) (2023) 19617–19641, http://dx.doi.org/10.1007/s11356-023-25148-9.

[34] L. Li, K. Jamieson, A. Rostamizadeh, E. Gonina, M. Hardt, B. Recht, A. Talwalkar, A system for massively parallel hyperparameter tuning, 5, 2020, pp. 1–17, http://dx.doi.org/10.48550/arXiv.1810.05934, arXiv:1810.05934.

[35] A. Dudáš, M.T. Udristioiu, T. Alkharusi, H. Yildizhan, S.K. Sampath, Examining effects of air pollution on photovoltaic systems via interpretable random forest model, Renew. Energy 232 (2024) 121066, http://dx.doi.org/10.1016/j.renene.2024.121066.

[36] Y. Wang, H.V. Thanh, H. Zhang, M. Rahimi, Z. Dai, L. Abualigah, Low-carbon advancement through cleaner production: A machine learning approach for enhanced hydrogen storage predictions in coal seams, Renew. Energy 241 (2025) 122342, http://dx.doi.org/10.1016/j.renene.2025.122342.

[37] F. Yuan, Z. Chen, Y. Liang, Precise solar radiation forecasting for sustainable energy integration: A hybrid CEEMD-SCM-GA-LGBM model for day-ahead power and hydrogen production, Renew. Energy 237 (2024) 121732, http://dx.doi.org/10.1016/j.renene.2024.121732.

[38] A. Alsharef, K. Aggarwal, Sonia, M. Kumar, A. Mishra, Review of ML and AutoML solutions to forecast time-series data, Arch. Comput. Methods Eng. 29 (7) (2022) 5297–5311, http://dx.doi.org/10.1007/s11831-022-09765-0.

[39] Z. Song, F. Xiao, Z. Chen, H. Madsen, Probabilistic ultra-short-term solar photovoltaic power forecasting using natural gradient boosting with attention-enhanced neural networks, Energy AI 20 (2025) 100496, http://dx.doi.org/10.1016/j.egyai.2025.100496.

[40] Z. Yu, R. Yu, X. Ge, J. Fu, Y. Hu, S. Chen, Tabular prior-data fitted network for urban air temperature inference and high temperature risk assessment, Sustain. Cities Soc. 128 (2025) 106484, http://dx.doi.org/10.1016/j.scs.2025.106484.

[41] J. Zhang, Y. Ge, Y. Wang, J. Tao, Z. Li, S. Fu, X. Wang, Y. Zhong, B. Yan, G. Chen, Photovoltaic power plants in mountainous area: Environmental impacts analysis based on random forest algorithm, Renew. Energy 254 (2025) 123670, http://dx.doi.org/10.1016/j.renene.2025.123670.

[42] S.F. Stefenon, L.O. Seman, E.C. da Silva, E.C. Finardi, L.d.S. Coelho, V.C. Mariani, Hypertuned wavelet convolutional neural network with long short-term memory for time series forecasting in hydroelectric power plants, Energy 313 (2024) 133918, http://dx.doi.org/10.1016/j.energy.2024.133918.

[43] S.F. Stefenon, L.O. Seman, C.K. Yamaguchi, L.D.S. Coelho, V.C. Mariani, J.P. Matos-Carvalho, V.R.Q. Leithardt, Neural hierarchical interpolation time series (NHITS) for reservoir level multi-horizon forecasting in hydroelectric power plants, IEEE Access 13 (2025) 54853–54865, http://dx.doi.org/10.1109/ACCESS.2025.3554446.

[44] S.M. Lundberg, S.I. Lee, A unified approach to interpreting model predictions, in: Conference on Neural Information Processing Systems, vol. 31, NIPS, Long Beach, CA, USA, 2017, pp. 4768–4777.

[45] T. Ho Thi Thanh, T. Huynh, H. Vu, B. Le, Shapley-based graph explanation in embedding space, Knowl.-Based Syst. 300 (2024) 112244, http://dx.doi.org/10.1016/j.knosys.2024.112244.

[46] V. Mizdrakovic, M. Kljajic, M. Zivkovic, N. Bacanin, L. Jovanovic, M. Deveci, W. Pedrycz, Forecasting bitcoin: Decomposition aided long short-term memory based time series modeling and its explanation with Shapley values, Knowl.-Based Syst. 299 (2024) 112026, http://dx.doi.org/10.1016/j.knosys.2024.112026.

[47] B. Han, G. Zou, W. Wang, J. Li, X. Zhang, Photovoltaic DC series arc fault detection method based on two-stage feature comprehensive decision, Sol. Energy 285 (2025) 113084, http://dx.doi.org/10.1016/j.solener.2024.113084.