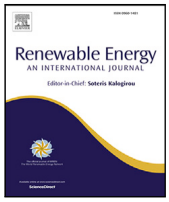
Renewable Energy 216 (2023) 118997

Contents lists available at ScienceDirect

Renewable Energy

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



Short-term photovoltaic power forecasting using meta-learning and numerical weather prediction independent Long Short-Term Memory models

Elissaios Sarmas [a,*], Evangelos Spiliotis [b], Efstathios Stamatopoulos [a], Vangelis Marinakis [a], Haris Doukas [a]

[a] Decision Support Systems Laboratory, School of Electrical & Computer Engineering, National Technical University of Athens, Greece
[b] Forecasting and Strategy Unit, School of Electrical & Computer Engineering, National Technical University of Athens, Greece

ARTICLE INFO

Keywords:
Photovoltaic power
Forecasting
Deep learning
Meta-learning
Long short-term memory neural networks

ABSTRACT

Short-term photovoltaic (PV) power forecasting is essential for integrating renewable energy sources into the grid as it provides accurate and timely information on the expected output of PV systems. Deep learning (DL) networks have shown promising results in this area, but depending on the weather conditions and the particularities of each PV system, different DL architectures may perform best. This paper proposes a meta-learning method to improve one-hour-ahead deterministic forecasts of PV systems by dynamically blending the base forecasts of multiple DL models to learn under what conditions each model performs best. Four base models of different long short-term memory architectures are used to produce PV production forecasts without using numerical weather predictions, with the objective to enhance the generalizability of the proposed solution. The accuracy of the meta-learner is evaluated using three rooftop PV systems in Lisbon, Portugal. Results indicate that different base models perform best at different PV plants, and meta-learning can improve accuracy by up to 5% over the most accurate base model per plant and up to 4.5% over the equal-weighted combination of the base forecasts. These improvements are statistically significant and even larger during peak production hours.

1. Introduction

Solar photovoltaic (PV) power generation, also known as PV production, involves the conversion of solar energy into electricity using solar panel technologies. These technologies provide a clean, silent, and low-cost form of electricity reducing carbon footprint and saving costs [1–3]. PV production has seen a rise of 22% in recent years, making it the second-largest absolute growth in renewable generation after wind [4]. However, operating PV plants can be challenging due to the high variability of PV production that stems from its correlation to weather conditions, such as solar irradiance and temperature [5]. This variability can pose difficulties in grid management as the penetration rate of solar energy increases [6], affecting electricity prices, the overall efficiency of energy markets, and the operating cost of local energy systems [7–9]. Short-term PV production forecasting (i.e. providing estimates on the expected PV production for the following hour(s) of the day) is critical for a number of reasons. Accurate PV forecasts are required for optimal economic load dispatch at the intra-hour and hour-ahead level, with the objective of minimally scheduling and combining power generating units and reducing electricity production cost [10,11]. On a smaller scale, short-term PV forecasting is also important for the efficient management of smart-grids and micro-grids, by making reliable dispatching plans for energy storage systems [12], optimizing load-shifting processes [13], reducing energy cost [14], managing reserves [15], and proactively maintaining the performance of PV plants [16]. As a result, the development of accurate short-term PV power forecasting models has become a hot area of research worldwide over the last decade [17,18].

A common feature of the majority of short-term PV power forecasting models proposed in the literature is that they depend on numerical weather predictions (NWPs; [19]). Such predictions can either be estimated using sophisticated irradiance forecasting models or purchased directly from weather services [20]. In the former case, the overall computational cost for developing and executing the PV production forecasting model is significantly increased. Even if the increased effort and complexity are not major issues for large-scale electricity producers (e.g. aggregators), the same is not necessarily true for small-scale PV owners (e.g. rooftop PV system operators), who seek accurate forecasts without the need to connect with meteorological services. In the latter case, where the successful execution of the PV power forecasting model

* Corresponding author.
E-mail address: esarmas@epu.ntua.gr (E. Sarmas).

https://doi.org/10.1016/j.renene.2023.118997
Received 23 November 2022; Received in revised form 28 June 2023; Accepted 5 July 2023
Available online 13 July 2023

Nomenclature

AE	Auto-encoder
ARIMA	Auto-Regressive Integrated Moving Average
ARMA	Auto-Regressive Moving Average
BiGRU	Bidirectional Gated Recurrent Unit
BiLSTM	Bidirectional Long Short-Term Memory
BPTT	Back Propagation Through Time
CLSTM	Convolutional Long Short-Term Memory
CNN	Convolutional Neural Networks
ConvGRU	Convolutional Gated Recurrent Unit
ConvLSTM	Convolutional Long Short-Term Memory
COR	Correlation Coefficient
CSO	Competitive Swarm Optimizer
CSRM	Clear Sky solar Radiation Model
DA	Direct Accuracy
DLNN	Deep Learning Neural Network
DNN	Deep Neural Network
EBP	Error Back Propagating
ElmanNN	Elman Neural Network
ELMNN	Extreme Learning Machine Neural Network
EU	European Union
FFNN	Feedforward Neural Network
GASVM	Genetic Algorithm-based Support Vector Machine
GRNN	Generalized Regression Neural Network
GRU	Gated Recurrent Unit
GWO	Grey Wolf Optimization
LGBM	Light Gradient Boosting Machine
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MARS	Multivariate Adaptive Regression Spline
MBE	Mean Bias Error
MHE	Mean Huber error
MLP	Multilayer Perceptron
MRE	Mean Relative Error
MSE	Mean Squared Error
MSLE	Mean Squared Logarithmic Error
NAR	Nonlinear Auto Regressive
NIA	Negative Index of Agreement
NN	Neural Network
NRMSE	Normalized Root Mean Squared Error
NWP	Numerical Weather Prediction
PDPP	Partial Daily Pattern Prediction
PHANN	Physical Hybrid Artificial Neural Network
PSF	Pattern Sequence-based Forecasting
PV	Photovoltaic
R2	Coefficient of Determination
RBFNN	Radial Basis Function Neural Network
RE	Renewable Energy
REL	Reliability Coefficient
RF	Random Forest
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SARIMA	Seasonal Auto-Regressive Integrated Moving Average
SARIMAX	Seasonal Auto-Regressive Integrated Moving Average Exogenous
SVM	Support Vector Machine
SVR	Support Vector Regression
TDNN	Time Delay Neural Network
U1	Theil U-statistic 1
U2	Theil U-statistic 2
VAR	Vector Autoregressive
WMAE	Weighted Mean Absolute Error
WNN	Wavelet Neural Network
WPD	Wavelet Packet Decomposition
WT	Wavelet Transform
WVCFM	Weight-Varying Combination Forecast Model
XGB	eXtreme Gradient Boosting

depends on external services, several risks may emerge (e.g. due to communication issues and possible changes to the format or the details of the provided NWPs; [21]). Either way, both approaches build on NWPs that, even within short periods of time, can vary significantly, being also difficult to accurately estimate in practice at local level [22]. Drawing from the above, developing short-term PV power forecasting models that require as input only information about the monitored generated power and weather conditions become particularly attractive [23], constituting a more generalizable, reliable, and faster to implement and run alternative for energy system operators, energy traders, energy managers, and many other stakeholders.

Nonetheless, even if short-term PV power forecasting was restricted to NWP independent models, there would still be numerous algorithms to chose from for developing a forecasting method [9]. In the forecasting literature, however, it is generally accepted that no model can consistently outperform all models available (for an encyclopedic review on forecast combinations please refer to section 2.6 of [24]) and PV production forecasting is not an exception [25]. In this regard, forecasters have been combining the forecasts of multiple models, each making different assumptions and having a different structure with the objective of providing more accurate forecasts overall. For instance, some models may perform better at peak production hours, others may be more accurate during less productive hours, while others may be more robust to unusual changes in weather conditions. The key idea behind forecast combinations is that the errors of different models will cancel each other, thus resulting in more accurate results than the individual forecasting approaches. Literally hundreds of schemes have been proposed for combining forecasts, ranging from simple combinations (e.g. mean, median or mode of forecasts) to more sophisticated ones (e.g. based on in-sample and out-of-sample forecasting errors, information criteria, Bayesian probability theory, and linear or nonlinear regression methods). More recently, meta-learning has attracted the attention of researchers as it allows the combination to be performed conditional to a set of external variables (e.g. hour of the day or weather conditions) that may affect the performance of the base forecasting models and, therefore, the optimal weights for combining the corresponding forecasts. As an example, Montero-Manso et al. [26] used meta-learning for assigning weights to the forecasts of nine different base forecasting models using time series features as input, achieving the second position in the M4 forecasting competition [27].

Inspired by the benefits of conditional forecast combination, in this paper we examine a meta-learning method for blending the forecasts of several novel deep learning (DL) models that are currently considered very effective for short-term PV power forecasting but may still fail to consistently provide the most accurate forecasts individuality for every single PV plant [9]. Moreover, in contrast to most of the existing meta-learning methods, we opt to directly compute the "optimal" forecasts

using a model-driven, regression-based approach instead of estimating the "optimal" combination weights, which would effectively suggest a linear combination of the base forecasts.

The objective of this manuscript is to make a significant contribution to the advancement of research in the field of PV production forecasting. Building upon the recognized advantages of conditional forecast combination, we propose a novel meta-learning method specifically designed to improve the accuracy of short-term PV power forecasts. Although various DL models have shown promising results individually for PV power forecasting, they often lack consistency in providing the most accurate predictions for every PV plant, considering their unique characteristics and operational conditions [9]. Our work addresses this limitation by exploring the integration of several state-of-the-art DL models through a meta-learning framework. Unlike conventional meta-learning methods that focus on estimating optimal combination weights, we adopt a distinctive approach. Our method directly computes the 'optimal' forecasts using a model-driven, regression-based approach. By doing so, we provide a more robust and interpretable solution that goes beyond a mere linear combination of base forecasts. This regression-based approach allows us to leverage the inherent strengths of each DL model and generate fused forecasts that capture the nuanced patterns and complex relationships present in PV power generation data. Our framework contributes to the existing literature by offering an innovative and effective methodology for PV power forecasting that outperforms traditional combination techniques and yields improved accuracy across diverse PV plant settings.

The contributions of our study are summarized as follows:

- We develop and optimize four long short-term memory (LSTM) neural network (NN) models of different architectures that can accurately forecast PV production for the next hour (one-hour-ahead deterministic forecasts). Since a major objective of our study is to provide a generalized forecasting framework that can facilitate planning in smart-grid and microgrid applications, the DL models are tasked to produce deterministic (point) forecasts instead of probabilistic ones, which are more relevant at grid level optimization problems.
- We focus on NWP independent PV forecasting approaches, determining the features to be used as input to the LSTM models accordingly. As a result, the constructed models are faster to run and more reliable compared to NWP dependent forecasting methods.
- We propose the use of a support vector regression (SVR) model that employs meta-learning to optimally blend the forecasts of the four LSTM models. The blending process is conditional to external variables that impact the performance of the base forecasting models, thus allowing for a dynamic ensembling. Moreover, in contrast to most meta-learning approaches, the final forecasts are computed directly by the meta-learner using a nonlinear regression method instead of being the product of a weighted, linear combination of the base forecasts.
- We evaluate the forecasting accuracy of the meta-learner using representative benchmarks and a data set that includes three rooftop PV plants located in Portugal.

The rest of the paper is organized as follows. Section 2 provides a literature review on short-term PV production forecasting. Section 3 describes the base forecasting models and the proposed meta-learning method. Section 4 presents an empirical evaluation of the proposed method, providing details on the data used, the accuracy measures and benchmarks considered, as well as a discussion on our results. Concluding remarks are provided in Section 5.

2. Overview and related work on PV power forecasting using DL networks

In this section, related research that has been conducted in the field of PV power forecasting is presented. The first part of the section provides an overview about the way PV power forecasting can be distinguished into certain subcategories based on key criteria, namely the forecasting horizon considered, the type of methods employed, and the variables used by the forecasting models as input. The second part focuses on deep learning forecasting methods and reviews different, indicative approaches and implementations.

2.1. Overview and categorization

PV power forecasting methods can be categorized based on several criteria such as the forecasting horizon (very short-term, short-term, medium-term, long-term), the nature of the method used (physical, data-driven, ML/DL), and their dependence on NWP (dependent, independent).

Regarding the forecasting horizon, which is defined as the future time span that the method generates predictions for, several categorizations have been proposed in the literature. The most widely adopted horizon-based categorization involves the classification of forecasting methods into short-, medium- and long-term ones [28]. However, according to Gupta and Singh [29] and Ahmed et al. [9], an extra category of very short-term PV power forecasting methods can be considered, leading to the following definitions:

- Very short-term forecasting methods, providing forecasts from few seconds to several minutes ahead. Such methods are exploited for the energy management of smart grids, e.g. for instant and efficient compensation of active and reactive power [5,30].
- Short-term forecast forecasting methods, providing forecasts from 1 h ahead to 1 week ahead. Such methods are suitable for applications at smart grid level, ensuring power unit commitment, scheduling, and dispatching [28], being also useful for dynamic pricing and predictive maintenance [31].
- Medium-term forecasting methods, providing forecasts from 1 month to 1 year ahead. Such methods are commonly used for optimal network planning and maintenance schedule [9].
- Long-term forecasting methods, providing forecasts from 1 year to several years ahead. Such methods can be beneficial in power transmission, distribution, and storage [32].

An additional factor in differentiating PV power forecasting approaches is the nature of the method or algorithm employed for generating forecasts and its reliance on actual measurements. Generally, two overarching categories of methods exist: physical methods and data-driven methods. Physical methods are based on detailed weather data and key design parameters of PV panels, and employ physical equations to forecast PV power [33], as successfully demonstrated by Zhao et al. [34], Zameer et al. [35], and Lei et al. [36]. In contrast, data-driven methods utilize data analytics techniques to identify patterns and correlations within weather and PV production data, and predict PV power accordingly [9]. Physical methods are considered more suitable for long-term forecasting, while data-driven methods are better suited for short-term forecasting. Data-driven methods can also be divided into purely statistical and machine learning (ML). An indicative example of statistical methods is presented by Sharadga et al. [37] who used a variety of time series forecasting models, such as ARMA, ARIMA, and SARIMA, to forecast PV production. More recently, Haider et al. [38] compared the accuracy of the SARIMAX and Prophet models in predicting solar irradiance with that of various NN-based models, concluding that the former are more suitable for longer forecasting horizons, while the latter for shorter ones. Similarly, Akhter et al. [39] discussed the advantages of an indicative set of ML models, such as NNs, decision trees, random forests, gradient boosting, and SVMs. It must be noted that except for physical and data-driven methods, there are also hybrid methods that combine characteristics of the two approaches. A comparison of physical, data-driven, and hybrid methods can be found in [40].

Another factor that can distinguish forecasting approaches is their dependence or not on NWPs. In addition to the NWP independent approaches that have been developed over time, and the one we propose, there are also NWP dependent models. These utilize weather forecasts with the objective to improve their forecasting accuracy, since PV production is strongly related to weather conditions. Such methods have been considered for instance by Ogliari et al. [41] and Dolara et al. [42] who used various NN models, such as physical hybrid artificial NNs (PHANNs), for day-ahead forecasting. A detailed comparison of ML PV power forecasting methods based on NWP has been conducted by Markovics and Mayer [43]. On the other hand, NWP independent methods do not require NWPs as input features, thus being faster and easier to implement, while remaining relatively accurate for short-term forecasting. Wang et al. [44] proposes an alternative terminology for this distinction, using the terms "direct" and "indirect" forecasting for NWP dependent and independent methods, respectively.

2.2. Deep learning methods

DL is a special field of ML that has demonstrated state-of-the-art performance in PV power forecasting, outperforming traditional machine learning and statistical models. This is due to the ability of said models to discover intricate structures in high-dimensional data [45] and produce more reliable results under uncertainty [46]. Over time, DL models have become more sophisticated, consisting of multilayered neurons, enabling them to capture more efficiently nonlinear and complex relationships between the input features and the output. In addition to the evolution in the type, number, and combination of cells used in DL architectures, the way the DL models learn is also evolving. For instance, Luo et al. [47] transfer learning and constrained long short-term memory are combined in a novel way to predict power generation of newly-constructed PV plants. Currently, most DL architectures involve convolutional NNs (CNNs), typically used for detecting patterns and structures in images, and recurrent NNs (RNNs), commonly used for predicting data with temporal relationships. Moreover, CNNs and RNNs are frequently combined with feedforward NNs (FFNN), using stacks of dense layers in their outputs, so that the forecasts of the NNs are properly tuned.

In Table 1 we present some recent DL methods used for short-term PV power forecasting. As seen in the first column of the table, there is no DL model that is systematically used over the others. Nonetheless, the most common implementations are based on RNNs, CNNs, or combinations of these two types of NNs. Regarding the input features, many studies focus on direct forecasting, using historical (lagged) values of PV power output combined with historical weather data, while others implement indirect forecasting methods based on NWP. In general, as shown in the third column of Table 1, NWPs are usually employed when the forecast horizon is more than one hour as the performance of NWP independent models drastically reduces after that point. In the following paragraphs, some of the most promising DL methods are presented, along with the most interesting results and insights gained. However, direct quantitative comparison among different forecasting methods is challenging in practice as their accuracy heavily depends on the quality and resolution of the historical data available [48].

In the PV power forecasting literature there has been an extensive use of LSTM networks, a special class of RNNs that incorporates nonlinear, data-dependent controls into the RNN cell and ensures that the gradient of the objective function with respect to the state signal neither vanishes nor explodes. Wen et al. [10] used a deep LSTM model to predict the hourly PV production at short-term horizons based on schedule, weather, and timescale variables with the objective to optimally dispatch the load of a community microgrid. The results showed that LSTMs were more accurate than other ML algorithms. Lee et al. [49] proposed a model that combines CNN and LSTM networks for day-ahead PV production forecasting using previous day's PV power observations. Similarly, Wang et al. [50] proposed a hybrid CNN and LSTM model and compared its performance over pure LSTM and CNN models. The results failed to indicate a superior forecasting approach as CNN was proven more accurate for longer forecasting horizons, while LSTM for certain forecast periods. Apart from LSTM, RNN have been examined in multiple other interpretations. Song et al. [51] divided PV power generation into four task categories (residential, agricultural, industrial and commercial) and proposed a multitasking RNN with the ability to transfer and reuse inter-task learning.

Another DL class of models that has recently been used for short-term PV power forecasting is the gated recurrent unit (GRU). GRUs are similar to LSTMs but involve fewer parameters, thus requiring less time to be trained [52]. Motivated by the potential of GRU models, Abdel-Basset et al. [53] and Rai et al. [54] proposed ways for redesigning these networks. Abdel-Basset et al. [53] modified the conventional GRU cell using dilated convolutions and residual connections to enable the network to capture the spatial–temporal representations of PV data and reported accuracy improvements over existing DL implementations, including the CNN and LSTM models of Wang et al. [50], the LSTM model of Sharadga et al. [37], and the GRU model of Lee and Kim [55]. Rai et al. [54] proposed a GRU-based auto-encoder and compared its performance over LSTM, CNN, and GRU models, as well as hybrids for the case of 24-hour, 48-hour, and 15-days forecast horizons, reporting promising results.

Regarding CNNs, their popularity in PV power forecasting can be mainly attributed to their ability to process data with grid topology features and their efficiently to extract hidden structures and inherent features [9]. Zang et al. [56] used popular CNN architectures, namely ResNet (residual network; [57]) and DenseNet (density convolutional network; [58]) for day-ahead PV production forecasting and compared their performance to statistical, natural, ML, and standard CNN models, reporting encouraging results. Similar conclusions were drawn by Aslam et al. [30] who performed an extensive comparative study and found DL and particularly hybrid DL models to achieve superior accuracy.

Finally, it has been shown that ensembling multiple models (aggregating the predictions of multiple independent base prediction models) has been very effective for short-term PV power forecasting according [9]. Ensembling may refer to combining different models or to combining different instances of the same model with different parameter settings. Some indicative studies using ensembling can be found in [59], where a simple average ensemble is used, and Liu et al. [60], where a weighted average ensemble model is employed. Except for average-based ensembling, some studies have considered meta-learning for PV power forecasting. Meta-learning refers to the process of using a ML (or DL) model for blending the baseline forecasts instead of using a conventional operator (e.g. mean, median, and mode). For instance, Lateko et al. [61] proposed a RNN-based meta-learner that combined the forecasts of five base models, namely ANN, DNN, SVR, LSTM, and CNN. Similarly, Eom et al. [32] proposed a CNN-based feature-selective meta-learner that blended the forecasts of two statistical models and a LSTM.

As evidenced by the large number of studies presented in Table 1, the upsurge of electricity production data from solar panels has led to the recognition of DL models as the most appropriate approach for PV power forecasting. Especially LSTM NNs can be exploited for multivariate forecasting, are capable of learning complex features from sequential data, and can effectively produce multi-step forecasts. However, as shown in this review section, current solutions have not yet exploited the full potential of DL and further research is required in the area of combining and comparing different DL architectures to further improve forecasting accuracy. Also, a limited number of studies have focused on creating ensemble models with the objective to capitalize on the combination of different DL architectures.

Table 1
Literature review on DL, short-term PV power forecasting methods.

Model	Features	Forecasting horizon	Measures	Source
NN – LSTM – XGB	Lagged PV Output - Forecasted global horizontal irradiance, temperature & relative humidity - Calendar Data	15-min – 1-h – 1-day ahead	MAE – RMSE – R2	Khan et al. [62]
AE – GRU	Lagged PV Output	1-hour ahead	MAE – MSE – R2	Rai et al. [54]
NN – Gradient Boosting	Forecasted irradiance, temperature, cloud coverage & wind speed	1-week ahead	RMSE – MAE	Sarmas et al. [48]
PVNET (ConvGRU – BiLSTM)	Lagged PV Output - Lagged wind direction, diffuse horizontal radiation, global horizontal radiation, weather relative humidity, weather temperature - Current Phase Average	5-min ahead	MAE – MAPE – RMSE	Abdel-Basset et al. [53]
RNN meta-learner, ANN, DNN, SVR, LSTM, CNN	Lagged PV Output, Lagged air temperature, relative humidity, average wind speed	1-day ahead	MRE, MAE, nRMSE, R2	Lateko et al. [61]
LSTM – BiLSTM – GRU – BiGRU – CNN	Lagged PV Output	1-5-30-60-min ahead	COR – MAE – MAPE – RMSE	Mellit et al. [18]
FFNN – GRU – LSTM	Lagged PV, Inverter Output - Lagged global horizontal irradiance, ambient-temperature, absolute-air-pressure, wind direction, wind speed, relative-humidity	(1–6)-h ahead	MAPE – NRMSE	Du Plessis et al. [63]
RNN – LSTM	Lagged PV Output	1-day ahead	RMSE	Liu et al. [64]
RNN – LSTM – Hybrid(CSO – GWO)	Lagged Solar Radiation	1-h - 1-day - 1-week ahead	MSE – MAPE – DA	Jayalakshmi et al. [65]
CNN	Lagged PV Output - Lagged global horizontal radiation, diffuse horizontal radiation - NWP	1-day ahead	MAE – MASE – MHE – MSE – MSLE – NIA – U1 – U2	Zang et al. [56]
RF meta-learner, FFNNs	Lagged PV Output, Lagged solar irradiance, ambient temperature, wind speed	1-hour ahead	nRMSE, nMAE	Wang et al. [66]
LSTM – RNN	Lagged PV Output	1-day ahead	COR – MAE – RMSE	Wang et al. [67]
WPD – LSTM	Lagged PV Output (Decomposed) - Lagged Weather Data	1-h ahead	MAPE – MBE – RMSE	Li et al. [68]
LSTM	Lagged PV Output - Lagged cloud coverage, visibility, temperature, dew point, relative humidity, wind speed	1-h ahead	MAE – MAPE – RMSE – R2	Mishra et al. [69]
RBFNN – CSO	Historical Weather Data	1-day ahead	RMSE – STD	Yang et al. [70]
NN – BI-LSTM – ARMA – ARIMA – SARIMA	Lagged PV Output	1–2-h ahead	R2 – RMSE	Sharadga et al. [37]
GASVM	Lagged PV Output - Lagged solar irradiance, temperature	1-hour ahead	MAPE – RMSE	VanDeventer et al. [71]
CNN	Lagged PV Output - Sky Images	15-min ahead	RMSE – Forecast Skill Index	Sun et al. [72]
LSTM – CNN – CLSTM	Lagged PV Output - Lagged wind speed, weather temperature, weather relative humidity, global horizontal radiation, diffuse horizontal radiation, wind direction - Current Phase Average	1-day ahead	MAE – MAPE – RMSE	Wang et al. [50]
Simple Averaging meta-learner, RNN mGRU, XGBoost, MLP	Lagged PV panel operating state parameters, Lagged Meteorological Parameters	1-day ahead	RMSE, MAE, MAPE	Zhu et al. [59]
RNN – BPTT	Lagged PV Output	15-30-90-min ahead	MAE – RMSE – MAPE – R2	Li et al. [73]
CNN(PVPNet) – LSTM	Lagged PV Output - Lagged temperature, solar radiance	1-day ahead	MAE – RMSE	Huang and Kuo [74]
DRNN-LSTM	Lagged temperature, humidity, wind speed, horizontal radiation, diffuse horizontal radiation	1-day ahead	RMSE – MAE – MAPE – PCC	Wen et al. [10]
Weighted Averaging meta-learner, SVM, MLP, MARS	Lagged PV Output, Lagged wind speed, weather temperature, weather relative humidity, global horizontal radiation, diffuse horizontal radiation and wind direction	1-day ahead	RMSE, MAE, MAPE	Liu et al. [60]
GRNN – ELMNN – ElmanNN	Lagged PV Output - Weather Clustering Analysis	5-min ahead	MAPE	Liu et al. [75]
NN(LM – LRNNA)	Lagged solar irradiation	1-day ahead	RMSE	Rodríguez et al. [76]
PSF – NN – DL	Lagged PV Output	1-day ahead	RMSE – MAE	Torres et al. [77]
CNN – LSTM	Lagged PV Output - Lagged temperature, wind speed, humidity, solar irradiation	1-day ahead	MAPE – RMSE – MAE	Lee et al. [49]
NN – CSRM - REL	Lagged PV Output - Forecasted Weather Data	1-day ahead	NMAE – WMAE – nRMSE	Leva et al. [78]
Deterministic – PHANN – EBP	Lagged PV Output - Forecasted Weather Data	1-day ahead	MBE – MAE – NMAW – R2	Ogliari et al. [41]
NN Ensemble – SVM	Lagged PV Output - Lagged solar irradiance, temperature, humidity, wind speed	5-min to 1-h ahead	MAE – MRE	Rana et al. [79]
WNN – WT	Lagged solar irradiance	1-h ahead	MBE – nRMSE	Sharma et al. [80]
PHANN (NN + CSRM)	Lagged PV Output - Forecasted Weather Data	1-day ahead	WMAE – NMAE – nRMSE	Dolara et al. [42]
DANN – TDNN – NAR	Lagged PV Output - Forecasted Weather Data	1-h ahead	RMSE – RE	Almonacid et al. [81]

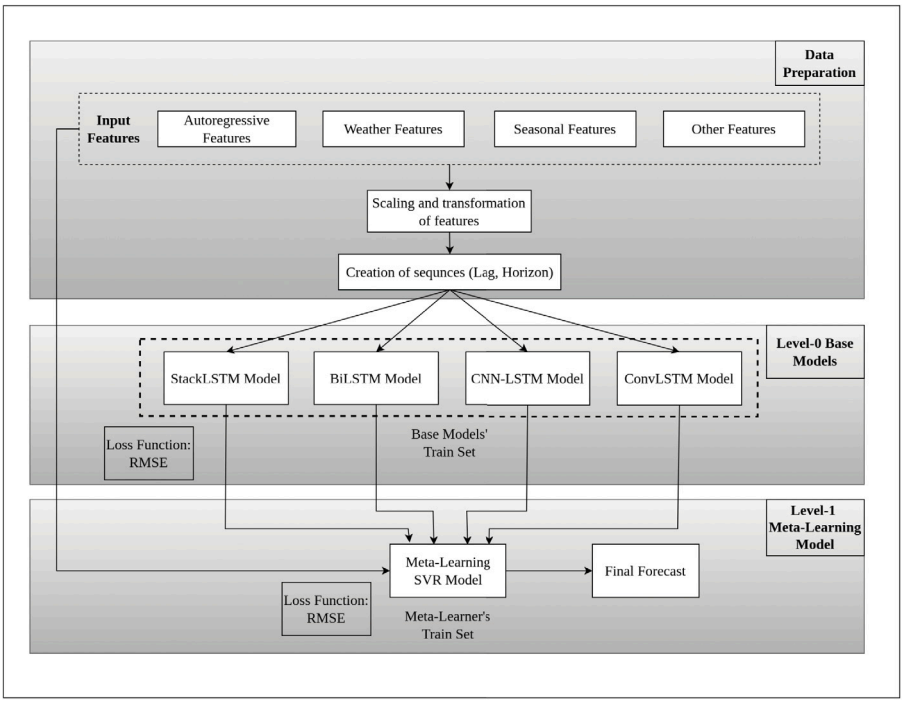


Fig. 1. The proposed meta-learning forecasting method.

3. Proposed method

In this section the proposed method is introduced. First, the overall methodological framework is presented. Then, a detailed description of the LSTM cell is provided, as it is the core component of all the NN architectures considered for developing the base forecasting models. Finally, the architectures of the base forecasting models and the meta-learner are described.

3.1. Framework overview

The proposed meta-learning method (Meta) consists of three individual phases: data preparation, training of the base forecasting models, and training of the meta-learner. The first phase involves a set of actions that must be performed in order to collect raw data and transform them to properly formatted, structured information that can be used as input, both by the base forecasting models and the meta-learner. The second phase includes the process of training the base forecasting models using the processed data as input and generating base (level-0) forecasts as output. The final phase consists of the training process of the meta-learner described in Section 3.4, which receives the base forecasts along with additional features as input and produces the final (level-1) forecasts as output. These phases are summarized in Fig. 1.

The data preparation phase includes two main steps. The first step refers to the selection of the features to be used as predictor variables in the base forecasting models described in Section 3.3 as well as the inputs to be used by the meta-learner for "optimally" blending the base forecasts. In short-term PV power forecasting settings this may involve the collection or engineering of autoregressive features (past PV production observations), weather features (e.g. past solar irradiance and temperature observations), seasonal features (e.g. hour and month), and other features that may contribute towards more accurate forecasts. Note that, although in principle the features of the base models and the meta-learner may differ, in our case are the same for the sake of simplicity. The second step refers to the processing of the raw data that facilitates the training of the base ML models, performed as described in Section 3.3. This may for instance include data cleansing, scaling, and transformation. More details on the data preparation phase of the proposed meta-learning method, tailored for the examined case study, are provided in Section 4.

The base models have been implemented using the Tensorflow Keras library for Python, version 2.6.0 [82], while the meta-learner has been implemented using the Scikit-learn library for Python, version 1.0 [83].

3.2. LSTM cell

Recurrent neural networks (RNNs) are a special type of NNs specifically designed to facilitate prediction of data with temporal relationships. In RNNs, the output of the network from a specific time step is passed as an input to the next time step, enabling the model to perform predictions based on the combination of the input of the current time step and the output of prior time steps. One of the most widely utilized RNNs is the LSTM network. LSTMs incorporate an internal memory unit that allows the model to effectively accumulate the state over the input features, making them particularly suitable for time series forecasting where correlations of unknown duration between the observations and important events may appear.

The LSTM architecture has been inspired by analyzing the error flow in conventional RNNs, resulting to the conclusion that long time dependencies could not be handled by such models [84,85]. The popularity of LSTMs derives exactly from their innate ability to overcome the most significant problems of common RNN architectures: vanishing gradients and exploding gradients. RNNs usually have a limited range of information that they can process and incorporate to their predictions. The main cause of this problem is that the influence (weighting factor) of the input on the network output may degrade exponentially, causing the vanishing gradient problem, or explode exponentially, causing the exploding gradient problem [86]. To deal with the above issues, LSTMs involve cells that incorporate an additive gradient structure, enabling a desired behavior from the error gradient on every step of the training process [87]. The architecture of the LSTM cell, as established by Graves [88] and Olah [89], is summarized in Fig. 2.

In contrast to conventional RNNs, which include a single unit, standard LSTM cells include four units. The most important concept

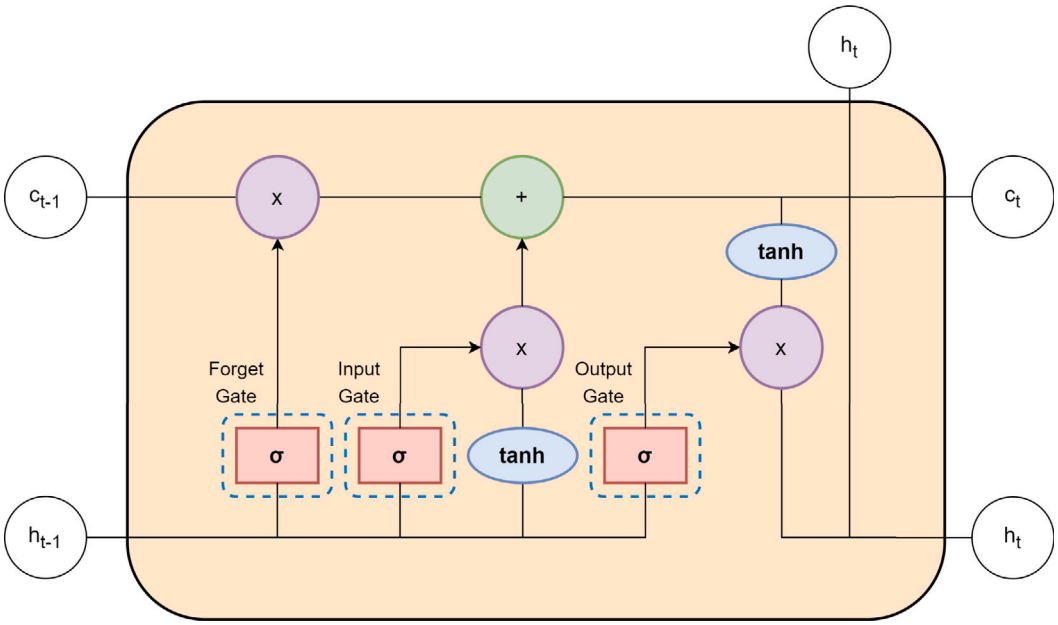


Fig. 2. Overview of the LSTM cell.

of the described architecture is the cell state depicted by the horizontal line at the top of Fig. 2. Every line in the figure can be considered as a vector to which several pointwise operations are performed. The cell state encapsulates the long-term memory capability of such networks storing and loading information of not necessarily immediately previous events. Apart from the cell state, LSTM cells also include the working memory capability that transfers information from immediately previous events, known as hidden state. The hidden state of the LSTM cells is overwritten at every step.

LSTM cells require three inputs and produce two outputs. The inputs, passed in vector form, are the following: the current input x_t, the previous hidden state h_{t-1}, and the previous cell state c_{t-1}. The core functionalities of an LSTM cell are implemented through its three gates: the forget gate, the input gate, and the output gate.

The forget gate is the first block represented in the LSTM architecture. The role of the forget gate is to determine which part of the information must be retained or discarded. The inputs of this gate are the previous hidden state h_{t-1} and the current input x_t. These inputs are passed through the sigmoid function σ_g which results in output values between 0 (no information passes through) and 1 (all the information passes through). The activation vector of the forget gate f_t is given as follows

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f). \tag{1}$$

The input gate serves as an input to update the cell status. Input gate functions in two parts. Firstly, the previous hidden state and the current input are passed into the second sigmoid function σ_g. Secondly, the same inputs are passed into the hyperbolic tangent function σ_c in order to regulate the network. Finally, the cell state vector c_t is the result of the element-wise product of the cell input activation vector $\tilde{c}_t$ and the update gate's activation vector i_t. The input gate's activation functions are given by the following set of equations:

$$\tilde{c}_t = \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \tag{2}$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \tag{3}$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \tilde{c}_t \tag{4}$$

Finally, the output gate determines what the next hidden state h_t provided by the cell should be. The hidden state includes information on previous inputs and is utilized for prediction. The previous hidden state h_{t-1} and the current input x_t are passed into the third sigmoid function σ_g. Then, the modified cell state is passed to the hyperbolic tangent function σ_h. These outputs are multiplied element-wise allowing the network to determine which information the hidden state should carry.

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \tag{5}$$

$$h_t = o_t \circ \sigma_h(c_t) \tag{6}$$

The parameters $W \in \mathbb{R}^{h \times d}$, $U \in \mathbb{R}^{h \times h}$ and $b \in \mathbb{R}^h$ of the above equations represent weight matrices and bias vector parameters, respectively, that are learned during the training process. Note that this description illustrates how a single LSTM cell operates.

3.3. Base forecasting models

The LSTM cell, which was thoroughly described in Section 3.2, has been extensively incorporated in many DNN architectures for making predictions in a variety of energy forecasting applications. An indicative, recent example can be found in the work of Pang et al. [90], which features a physics-informed NN composed of bidirectional LSTMs accompanied by a Bayesian optimization algorithm. The aim of this NN was to accurately estimate heat generation of lithium-ion batteries under various driving conditions while cost-effectively optimizing the NN's hyper-parameters. Similarly, an innovative anti-noise adaptive LSTM with dual feedback correction has been proposed by Wang et al. [91] for predicting the remaining useful life of lithium-ion batteries. While both above-mentioned studies have featured innovative NN architectures, in the field of short-term PV power forecasting the most popular architectures are the following: Stacked LSTM, Bidirectional LSTM, CNN-LSTM, and ConvLSTM. Therefore, in this section we focus on these architectures and provide a detailed description of each model.

Note that although the four models are described in a relatively generic way, their hyper-parameters (namely the number of layers, the number of LSTM cells per layer, and the number of filters for CNNs) have been fine-tuned by performing grid search optimization. Similarly, although the inputs of the models are discussed in an abstract way, in Section 4.3 we clarify that they consist of tabular data, including solar irradiance variables, features that simulate the seasonal and calendar variations of the PV production, past observations of PV production, and categorical variable that indicate the overall profile of the weather.

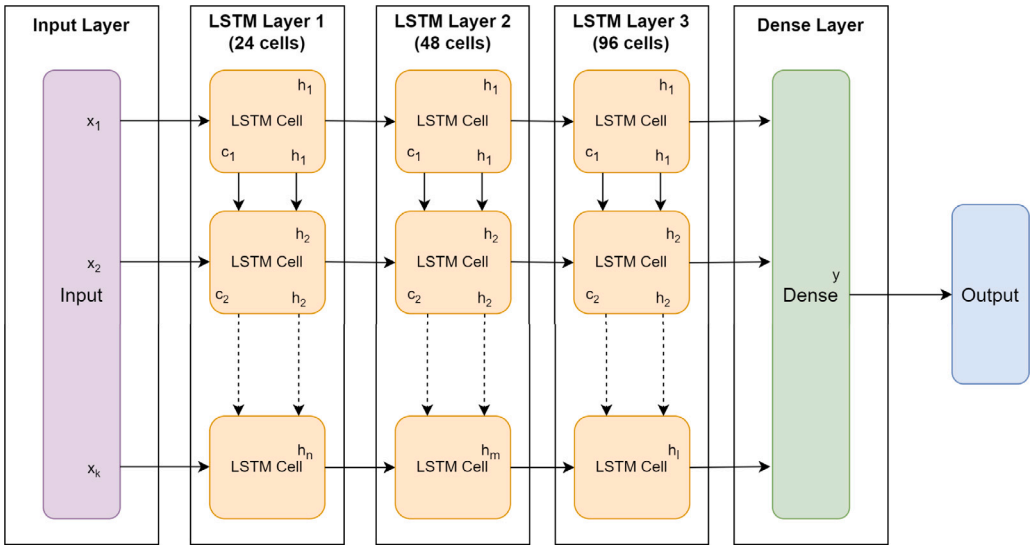


Fig. 3. The selected architecture of the stacked LSTM (StackLSTM) model.

3.3.1. Stacked LSTM

Standard architectures of LSTM models include a single fully connected layer of LSTM cells, followed by a feedforward output layer. The stacked LSTM (StackLSTM) model used in this study is composed of multiple fully connected layers, each including multiple LSTM cells. These layers are stacked sequentially, producing a deeper architecture that can deal with complex learning tasks more effectively [92]. When it comes to feedforward NNs, stacking multiple fully connected layers results in a hierarchical feature representation of the data that is used for training the model. The same applies to the case of deep LSTM models. Using multiple levels of fully connected LSTM layers enables the model to combine the representations learnt from previous layers and capture new patterns, adding a higher level of abstraction. According to Hermans and Schrauwen [93], each layer solves some part of the forecasting problem and passes the generated prediction to the next layer. Consequently, the architecture can be considered as a processing pipeline where each layer contributes to a different part of the problem-solving process until the last layer provides the final forecast. Therefore, the layers provide the model a strong capability of identifying more complex patterns and structures in the input time series.

In the examined model, each LSTM layer outputs a sequence of vectors which are directly passed as an input to the subsequent LSTM layer. A single LSTM layer takes a three-dimensional input and generates a two-dimensional output, including the hidden state of the last time step which interprets the end of the sequence. The next layer of the model, though, should be fed with a three-dimensional input too. Thus, the LSTM output of the prior layer, including the hidden state of all previous time steps, is passed to the subsequent one. This architecture enables all layers to have a three-dimensional input from the previous hidden layer to the subsequent one. This hierarchy of hidden layers enables more complex data representation of different scales to be learned [94].

The StackLSTM architecture used in this study is presented in Fig. 3. Let us define *batch_size* as the number of training data samples before the model's internal parameters are updated, *num_timesteps* as the number of previous time-steps given to the model and *num_features* as the length of the feature sequence (number of model's features). When fine-tuning the model, two hyperparameters were considered: the number of layers with possible values 1, 2, and 3 layers, and the size of each layer with possible values 24, 48, 96, and 192. The predominant architecture is comprised of 3 LSTM layers of 24, 48, and 96 cells, respectively.

3.3.2. Bidirectional LSTM

In many forecasting problems, it is beneficial to develop LSTM models that can learn temporal relationships in both directions, i.e. forward (past to future) and backwards (future to past). The main advantage of this approach is that by concatenating both interpretations, predictions are made using sequential information from all data points before and after them [95]. These models are generally called bidirectional recurrent neural networks [96] or, when the layers are composed of LSTM cells, bidirectional LSTMs (BiLSTMs).

The difference between BiLSTMs and standard LSTMs in terms of architecture lies in the fact that BiLSTMs incorporate two parallel layers in both propagation directions [97]. Thus, the forward passing and the backward passing of each layer are performed in a similar fashion to regular NNs, differing in the ability of these two layers to memorize information from both directions [98]. This feature of BiLSTMs adds a different approach as it bypasses the shortcoming of conventional LSTMs, which are only able to make use of the previous context. Therefore, BiLSTMs have been successfully used in a wide range of prediction tasks such as speech recognition [99] and sentiment analysis [100], among others, rendering them one of the most suitable models for time series forecasting.

The fine-tuning of the BiLSTM model was performed based on the number of bidirectional LSTM layers and the number of cells per layer. Regarding the number of layers, 1, 2 and 3 layers have been tested, while the number of possible cells per layer ranged in 24, 48, 96, and 192. The selected architecture, presented in Fig. 4, includes 1 bidirectional LSTM layer with 192 cells.

3.3.3. CNN-LSTM

Another approach for time series forecasting involves a customized architecture that combines CNNs with LSTMs. In general, CNNs are a special class of NNs, most commonly applied for visual imagery analysis, facial recognition, image search, and augmented reality [101]. CNNs can be thought of as regularized versions of multilayer perceptrons, which aim to exploit the hierarchical pattern in data, creating patterns of increasing complexity by using smaller and simpler patterns through filters.

The use of convolutional layers gives the model the distinct advantage to learn the internal representation and patterns of sequence data. The use of 1-dimensional CNNs has proven to be very efficient at learning features from input sequences. Moreover, it has been shown that pairing CNNs with LSTMs effectively combines the advantages of both NN architectures [102]. The combination of convolutional layers

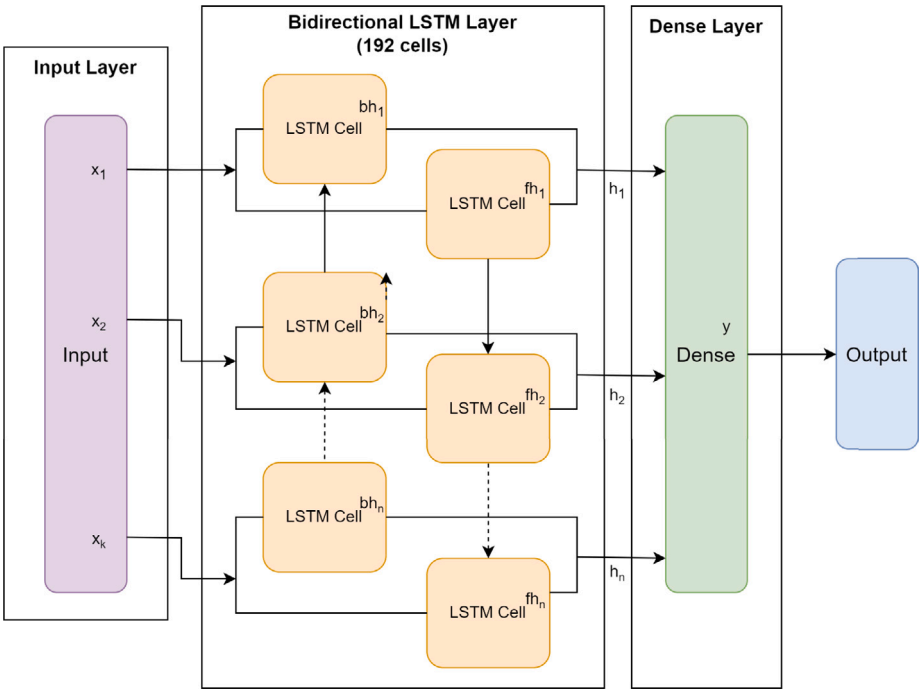


Fig. 4. The selected architecture of the bidirectional LSTM (BiLSTM) model.

and LSTM layers results in the creation of a hybrid model, called CNN-LSTM, where the convolutional layers are used to extract data patterns and the LSTM layers are exploited for making predictions. Effectively, each convolutional layer serves as an encoder that learns the structure of small sub-sequences of input data that are then fed as time steps to an LSTM layer.

The architecture of the CNN-LSTM model is composed of two parts. The first part is composed of convolutional and pooling layers that recognize and extract features from the input data, serving as pre-processing layers that feed information to fully connected network layers [103]. The convolutional layer is usually followed by a nonlinear activation function and a max pooling layer. The pooling layer performs a sub-sampling technique which extracts certain values from the convolved features and produces a lower dimension matrix. Max pooling is performed in a similar fashion to convolution, using a sliding window that takes as input the values of each patch of the convolved features and outputs the maximum of each patch's values [103]. The second part of the architecture consists of fully connected LSTM layers that learn long time dependencies of features leading to the dense layer which outputs the forecast.

From a technical point of view, it is necessary to split the input sequences into 2-dimensional sub-sequences in order to be processed by the CNN. Assuming that time series data are comprised of k time steps as input and one output, each input sample is split into $k/2$ subsequences, each composed of two time steps, enabling the CNN to interpret each subsequence separately giving this interpretation as input to the LSTM model. The fine-tuned parameters of the model were the number of filters (possible values: 32, 64, 128; selected value: 64), the number of LSTM layers (possible values: 1, 2, 3; selected value: 2), and the number of LSTM cells per layer (possible values: 25, 50, 100; selected values: 200 for layer 1 and 100 for layer 2). The architecture of the developed CNN-LSTM model is presented in Fig. 5.

3.3.4. ConvLSTM

In the CNN-LSTM architecture, CNN deploys convolutional operations attempting to capture spatial features and LSTM is responsible for learning sequential characteristics through its memory units. Convolutional LSTM (ConvLSTM), then, succeeds in integrating the advantages and capabilities of the CNN-LSTM architecture into convLSTM cells, offering a more compact implementation. The ConvLSTM model architecture is based on providing convolutions in the LSTM layers for the hidden and cell states, differing from the traditional CNN-LSTM that uses matrix multiplication [104]. This custom architecture facilitates the convolutional reading of input to be developed directly into each LSTM unit. In general, the ConvLSTM model is designed for taking spatial–temporal data in two dimensions as input. However, its architecture can also be used for multivariate time series forecasting.

Similar to the CNN-LSTM model, in ConvLSTM the time series has to be pre-processed before provided as input to the model. Thus, each input sample (comprised of k steps) is split into $k/2$ subsequences, each composed of two time steps. As a result, the model receives the input data in the format of k time steps in a two-dimensional representation composed of $k/2$ rows and two columns. The only parameter that has been fine-tuned for the developed ConvLSTM model is the number of filters for the convolutional LSTM layer. The possible values were 16, 32, 64 and 128, while the predominant one was 64. The selected architecture of the ConvLSTM model is summarized in Fig. 6.

To summarize the information presented above, it is evident that each of the four base forecasting models possesses distinct advantages and drawbacks in the realm of short-term PV power forecasting. Notably, the Stacked-LSTM model excels in capturing long-term dependencies, making it well-suited for sequential data and time series forecasting tasks. On the other hand, the Bi-LSTM model effectively handles incomplete data through its flexible connection mechanism, lending itself to multi-dimensional problems but potentially performing less optimally for longer forecasting horizons [105]. The CNN-LSTM model demonstrates proficiency in accurately processing sequential data while utilizing memory more efficiently, enabling the extraction of short-time local features from time series data [49]. Conversely, the ConvLSTM model excels in capturing long-time horizons and can be easily adapted for univariate time series forecasting, effectively modeling temporal feature dependencies and locally distributed relations and features [106]. Considering the specific characteristics and contexts of each model, it becomes apparent that certain models may outperform others in particular conditions. Therefore, no single model is likely to achieve optimal performance in all situations. This observation

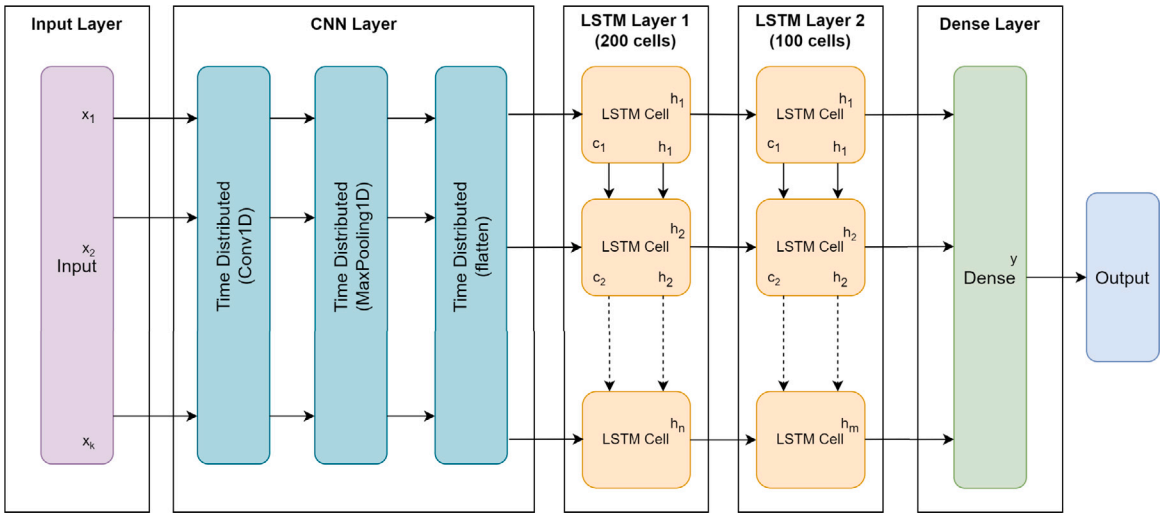


Fig. 5. The selected architecture of the CNN-LSTM model.

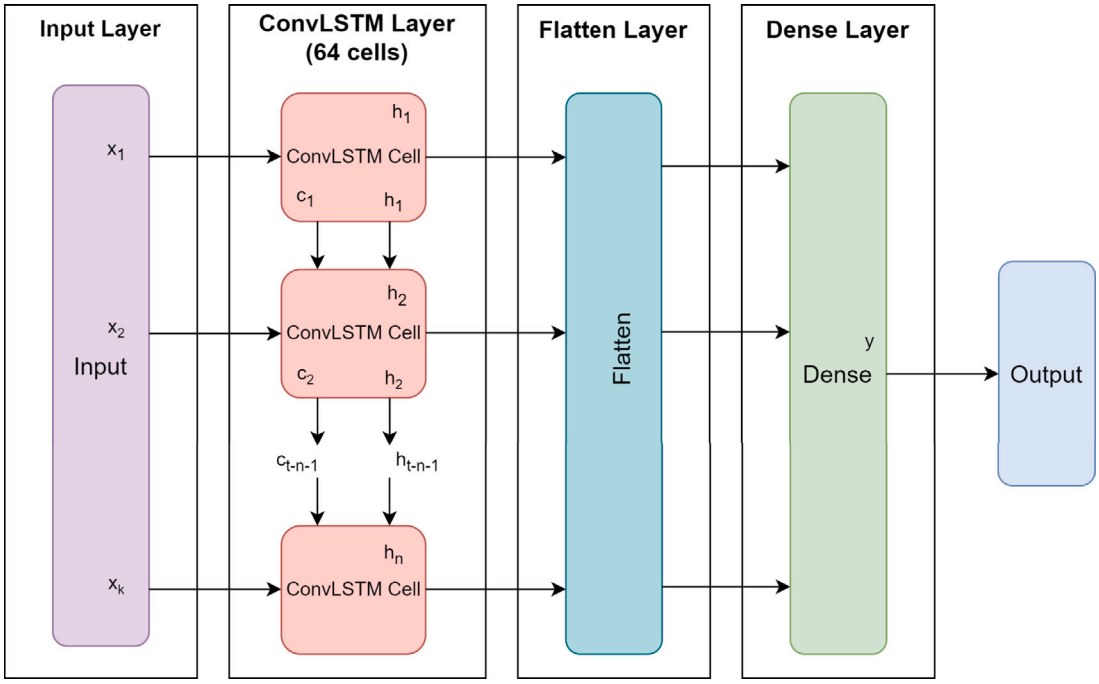


Fig. 6. The selected architecture of the ConvLSTM model.

motivates the adoption of a meta-learner, as proposed in Section 3.4, which selectively combines the capabilities of the four base models to produce more accurate and robust forecasting results overall. By leveraging the unique strengths and mitigating the drawbacks of each model, the meta-learner provides a promising avenue for improved forecasting outcomes.

3.4. Meta-learning

Meta-learning (or stacking) is the process of "learning to learn" in the sense that it assigns a ML model with the task to improve the accuracy of a specific forecasting task using as a base (meta-data) the forecasts of other ML forecasting models [107]. This task can be proven particularly helpful in prediction tasks where different forecasting models of diverse characteristics, structures, and data assumptions are expected to perform differently under various circumstances [108]. For example, in PV power forecasting some models may perform better in summer when weather conditions become more stable but under-perform in winter. Similarly, the accuracy of some models may increase during peak hours of PV production but deteriorate during low production hourly intervals.

Although the final forecasts (level-1 forecasts) produced by the meta-learner may effectively be a combination of the base forecasts (level-0 forecasts), either linear or nonlinear, meta-learning should not be seen as a conventional forecast combination scheme. This is because the objective of the meta-learner is to blend the base forecasts not by simply observing the average relative performance of the base models but by understanding the specific conditions under which a single base model or a combination of base models performs best [109]. If these conditions are learned, then forecast combination can be preformed in an effective, data-driven fashion that enables better generalization [110]. According to Hospedales et al. [111], meta-learning provides an opportunity to tackle many challenges of

conventional ML, including data and computation bottlenecks that are common for instance when working with DL NNs.

The meta-learner is implemented using the SVR method [112]. SVR is a widely used ML algorithm that performs regression in a similar fashion to conventional linear regression in the sense that the line used for making predictions is estimated as

$$\hat{y} = \mathbf{W}^T\mathbf{x} + b, \tag{7}$$

where $\hat{y}$ is the target variable, $\mathbf{x}$ is the set of features used as predictor variables, $\mathbf{W}$ is a vector of weights, and b is an intercept that accounts for bias. However, SVR uses a different way for calculating the prediction line, referred to as a hyperplane. The data on each side of the hyperplane which has the smallest distance to the hyperplane are called support vectors. Traditional regression methods aim to minimize the error between the actual values and the forecasts. On the opposite, the SVR method aims to fit the optimal line within a threshold value d which describes the distance between the hyperplane and the support vectors. Thus, in general, the SVR method satisfies the condition $-d < y - \mathbf{W}^T\mathbf{x} + b < d$, using the points within the boundary to generate forecasts. In this study, the radial basis function (RBF) is utilized as the kernel function of the SVR in order to map the sample data onto a high-dimensional feature space and allow for nonlinear regression.

SVR was preferred as a meta-learner over other ML models since it can combine the forecasts of the base models in a weighted fashion, while also capturing nonlinear relationships among features. Other popular options to be used as meta-learning models would be linear regression, tree-based regression, and NNs. However, all the aforementioned alternatives lacked in one or more characteristics compared to SVR. More specifically, although linear regression is a weighted, intuitive method it cannot capture nonlinear relationships among the selected features. Tree-based models, such as gradient boosted trees and random forests, cannot combine directly the level-0 with the level-1 forecasts, as the final prediction of such models is effectively the mean of the samples that satisfy the rules of the respective nodes. Finally, although NNs are appropriate for nonlinear regression, they typically require much larger train sets than SVR to be trained effectively. In addition to the above, since the final forecasts of the proposed meta-learner are computed by a nonlinear regression method, the blending mechanism examined is more flexible and generic compared to standard meta-learners that estimate the "optimal" combination weights of the base forecasts and employ a weighted average of those.

SVR was chosen as the preferred meta-learner over other ML models due to its distinctive advantages in combining base model forecasts in a weighted manner while capturing nonlinear relationships among features. Although alternative options such as linear regression, tree-based regression, and NNs are popular choices for meta-learning, each falls short in various aspects compared to SVR. Linear regression, although intuitive and weighted, lacks the ability to capture nonlinear relationships among selected features. Tree-based models, including gradient boosted trees and random forests, are unable to directly combine level-0 and level-1 forecasts, as their final prediction effectively represents the mean of samples satisfying respective node rules. In contrast, while NNs are suitable for nonlinear regression, they typically require significantly larger training sets than SVR to achieve effective training. Furthermore, the proposed meta-learner's adoption of a nonlinear regression method for computing final forecasts offers enhanced flexibility and generality compared to standard meta-learners that estimate "optimal" combination weights and employ a weighted average of base forecasts. This characteristic further accentuates the advantage of the blending mechanism examined in this study.

4. Case study

This section provides details on the case study examined for evaluating the forecasting performance of the proposed meta-learning method. This includes the description of the data set used, the experimental design, the measures selected for assessing forecasting accuracy, and the forecasting methods considered as benchmarks. The results are presented and discussed at the last part of the section.

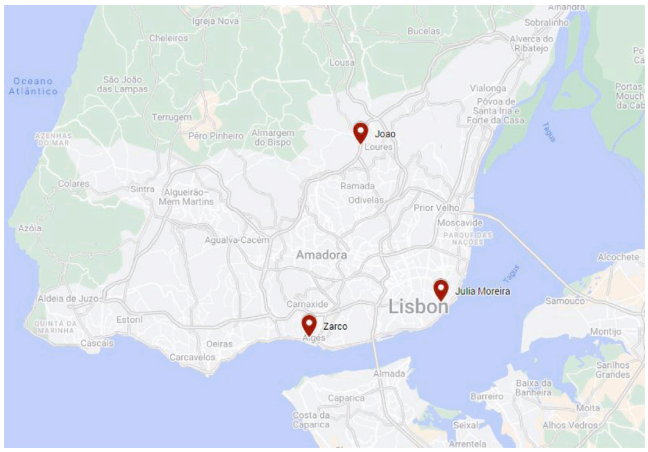


Fig. 7. Location of the three rooftop PV plans examined in the case study.

4.1. Data set

The data set includes three rooftop PV plants located in Lisbon, Portugal, namely "Joao", "Julia Moreira", and "Zarco". The locations of the plants are shown in Fig. 7. The three plants are installed in public buildings (institution of social solidarity, dental medicine unit, and elementary school) and have a peak power of 40, 27, and 50 kW, respectively.

The data used are hourly and have a duration of about 2 years and six months (21,958 h), ranging from 1 August 2018 to 31 January 2021. Fig. 8 presents the hourly distribution of the capacity factor for the three PV plants. As expected, electricity production is typically higher between 12:00 and 15:00, i.e. when solar irradiance is generally greater within the day, and significantly lower in the early morning hours (09:00 to 11:00) and in the afternoon (16:00 to 18:00), being zero otherwise (no solar irradiance in the nighttime). Nevertheless, it is clear that, depending on the weather conditions, PV production can vary significantly between different days. Moreover, we observe that the distributions of the examined parks are similar as the plants are located in close areas and have comparable orientations. Therefore, in the rest of the present subsection our analysis will focus on the "Joao" subset of data.

The strong daily seasonality of the data across consecutive days, that are naturally characterized by similar weather conditions, is visualized in the autocorrelation plot of Fig. 9. As seen, previous observations can become particularly useful features for developing forecasting methods, especially when used for forecasting PV production at the following few hours (e.g. lag = 1 to lag = 6). Fig. 9 also indicates that the month of the year could be another useful predictor variable as different months generally suggest different weather conditions and, as a result, PV production.

Apart from electricity production records, measured in kW, for each park, the data set includes the observed weather conditions in terms of solar irradiance (measured in W/m^2), diffuse solar irradiance (measured in W/m^2), and temperature (measured in °C). The weather variables are measured on site by weather stations installed in the PV parks and represent the average weather conditions of the corresponding hours.

Fig. 10 illustrates the distributions (histograms) of the electricity produced in the "Joao" PV plant and the three weather variables available as well as their pairwise correlations. We observe that PV production is strongly correlated to solar irradiance and diffuse solar irradiance, being also positively affected by temperature. Therefore, we conclude that all weather variables could be useful inputs for forecasting PV production.

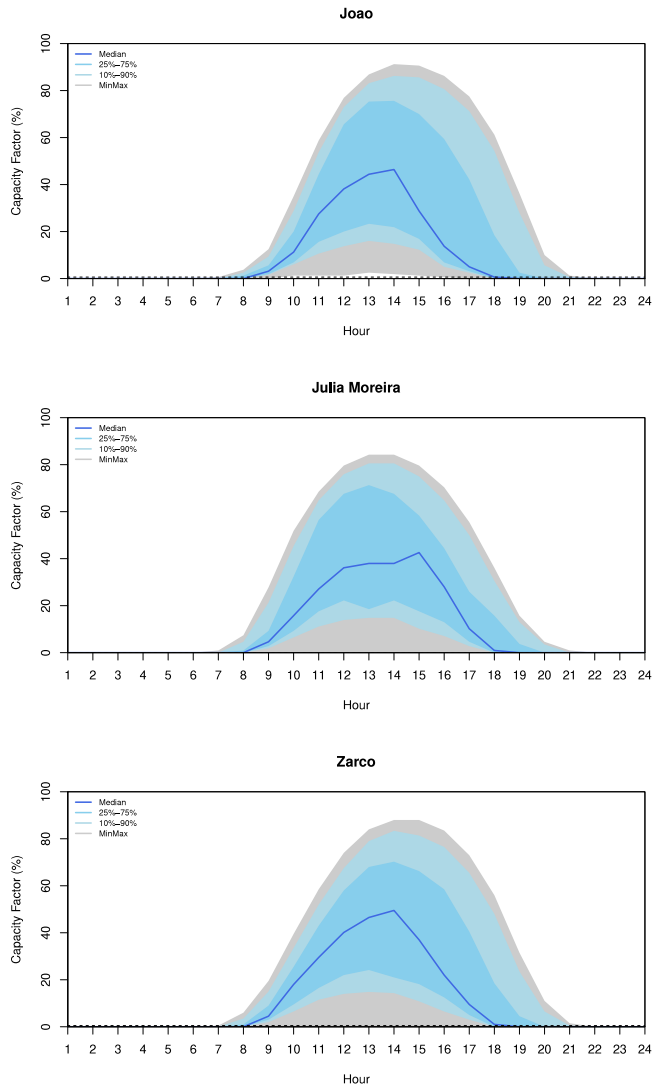


Fig. 8. Hourly distribution of the capacity factor (computed using an additive decomposition with moving averages; [113]) for the examined PV plants. Low variance around the seasonal profile (blue line) suggests strong daily seasonality (observations overlap every 24 h).

4.2. Experimental setup

In order for our results to be representative, capturing the dynamics of multiple time periods and, as a result, different weather conditions, we implement the rolling origin evaluation approach [114]. According to this approach, which is equivalent to cross-validation for time series data, a period of historical data is first used for training the forecasting method of interest. Then, the method is used for forecasting PV production for the following hour (one-step-ahead forecast) and, finally, the accuracy of the forecast is measured given the realized production and a performance measure of preference. Subsequently, the forecast origin is shifted by one hour and the same method is used for producing a new forecast, contributing another evaluation. This process is repeated till there are no data left for performing an evaluation and the overall performance of the method is determined based on its average accuracy over the performed evaluations.

In this case study we used the first year of data (8760 observations) for training the base forecasting models. Then, we used the trained models to produce forecasts for the following year, assembling a sufficiently large train set for the meta-learner. Consequently, we trained the meta-learner and tested its performance (along that of the base forecasting models) using a test set that consists of about 6 months of data in a rolling origin fashion. An overview of the experimental setup is provided in Fig. 11.

In terms of performance measures we consider the root mean squared error (RMSE), as follows

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2},$$

where y_t is the actual electricity produced at point t, $\hat{y}_t$ the corresponding forecast of the method being assessed, and n the number of forecasts produced when applying the rolling origin evaluation. Lower RMSE values suggest better accuracy. Given that the mean minimizes the sum of the squares of the forecast errors [115], the selected measure becomes particularly useful for identifying models that provide better forecasts, both on average, but also during the peak hours of production which are more critical for energy producers.

However, the accuracy of solar forecasting or PV power forecasting models should be inter-comparable across PV systems located in different locations or with measurements taken in different seasons [116]. Thus, the forecast skill index is used in addition to the RMSE measure for evaluating the relevant performance of the forecasting models [117]. This metric relies on comparing the proposed model to a reference model, using RMSE as a baseline accuracy measure. As a reference model, we select the persistence model which assumes that the future condition (forecast) will be the same as the present condition (real value) [118]. The persistence model is the most common reference model used in the literature for standardizing the verification of solar forecasting models. The forecast skill index is defined as follows:

$$\text{Forecast Skill Index} = 1 - \frac{\text{RMSE}_{proposed}}{\text{RMSE}_{reference}} \tag{8}$$

where $RMSE_{proposed}$ and $RMSE_{reference}$ are the accuracy estimates of the proposed and persistence forecasting models according to the RMSE measure, respectively.

4.3. Forecasting methods

We evaluate the forecasting accuracy of the four base models and the meta-learner described in Section 3. Undoubtedly, the effective implementation of the proposed methods depends on the selection of appropriate features that provide useful information as input to the respective forecasting algorithms. Based on the insights of Section 4.1, and given that we are interested in developing NWP independent models, we consider the following set of features:

- Solar irradiance measured in the last six hours by the weather stations.
- Features that simulate the seasonal and calendar variations of the PV production, namely the month of the year and the hour of the day. The former feature is transformed using one-hot encoding, while the latter using the sine and cosine transformations. Similar to solar irradiance, seasonal and calendar features are extracted for the last six hours.
- Past observations of PV production (lag = 1 to lag = 6) accounting for possible autocorrelations.
- A categorical variable indicating the weather profile of the last six hours. To determine the weather profile of each hour, the observations of the data set are clustered considering solar irradiance, diffuse solar irradiance, and temperature as input to the k-means algorithm with the objective of contracting three clusters. The output of the algorithm is transformed using one-hot encoding, resulting in a total of three binary features. Table 2 illustrate the type of information the clusters provide to the forecasting methods. As seen, the first two clusters mostly correspond to hours that are characterized by high and medium solar irradiance, respectively, while the third cluster to hours of low irradiance and temperature.

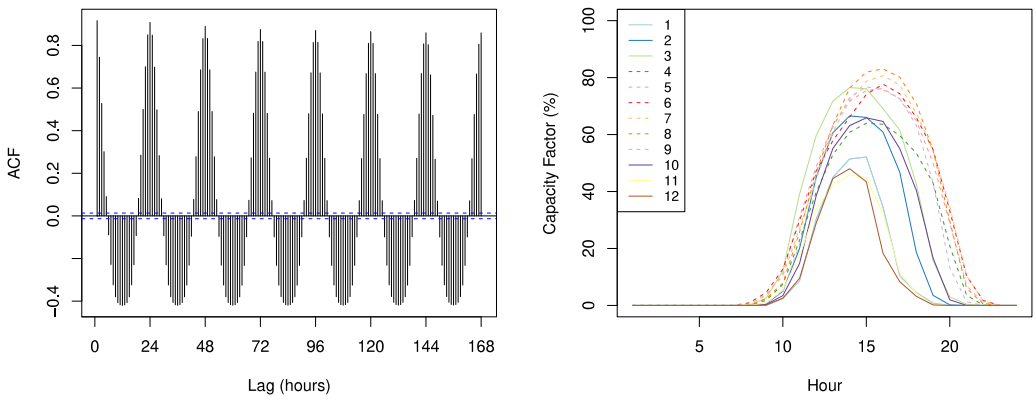


Fig. 9. Autocorrelation function of the PV production in the "Joao" plant (left) computed across the week (168 h) and average daily capacity factor (right) for each month of the year.

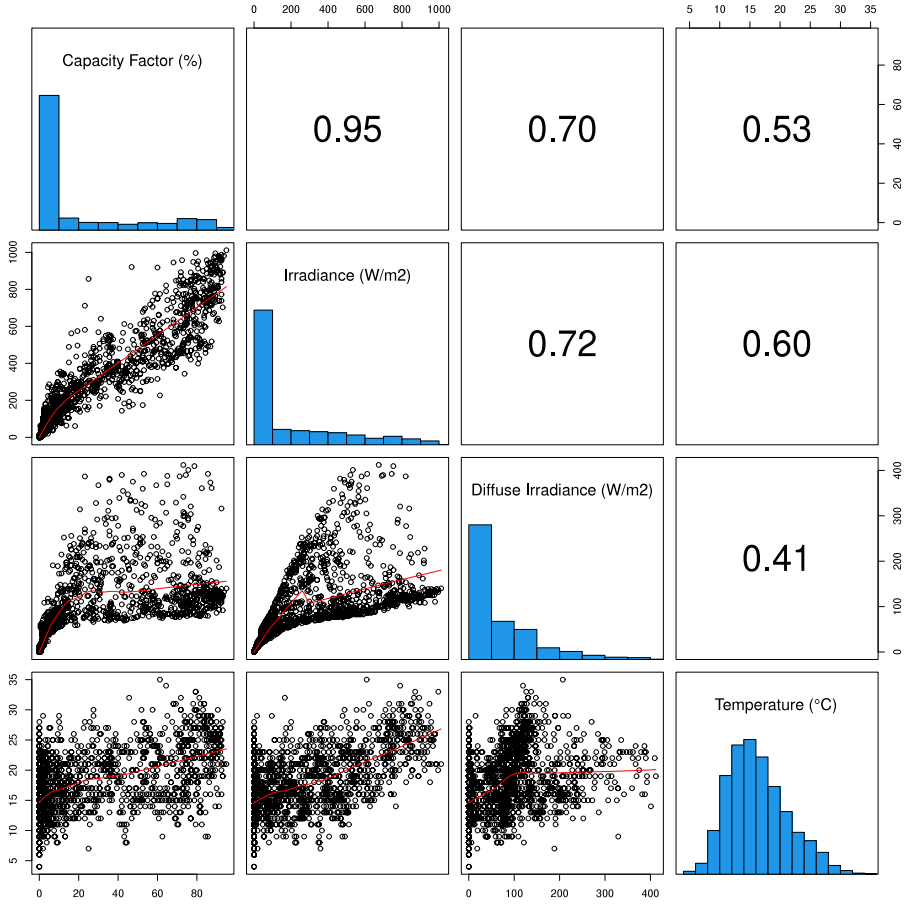


Fig. 10. Pairwise correlation of the electricity produced in the "Joao" PV plant (capacity factor) and the three weather variables available. The upper part of the triangular matrix presents the Pearson correlation coefficients of the variables while its lower part the respective scatterplots. The main diagonal demonstrates the distributions (histograms) of the variables.

Table 2
Weather profiles created for the "Joao" PV plant. For each of the three profiles the average and the standard deviation (in parenthesis) of the variables used for creating the clusters are reported.

Cluster	Irradiance	Diffuse irradiance	Temperature
1	709.95 (138.26)	168.75 (84.87)	23.34 (4.49)
2	316.18 (117.30)	129.78 (59.36)	17.76 (4.11)
3	13.13 (33.69)	8.01 (19.06)	14.94 (3.88)

Overall, the forecasting methods involve 19 features × 6 lags = 114 inputs that are scaled in a range between zero to one to facilitate training and allow better generalization.

4.4. Benchmarks

In order for the meta-learner to contribute towards more accurate forecasting, its predictions should outperform those of the four base models used for constructing the final forecasts. Effectively, such a finding would suggest that no individual base model is more accurate on average than the "optimal" forecasts computed by the meta-learning

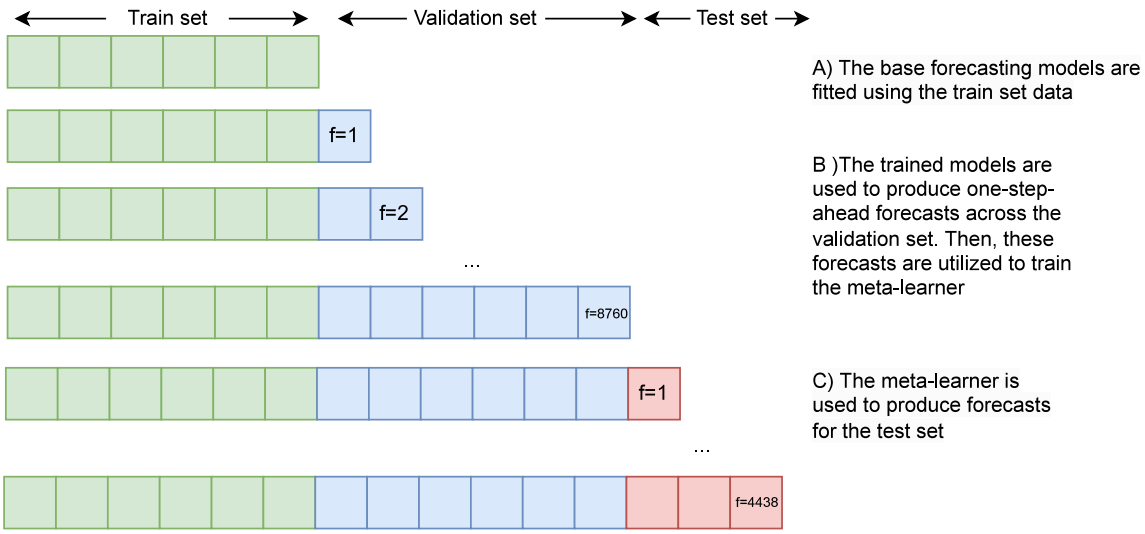


Fig. 11. Overview of the experimental setup considered in the case study. The train set (green) is first used in order for the base forecasting models to be fitted. Then, one-step-ahead forecasts are produced by all base models across the validation set (blue) using the rolling origin approach. Finally, these forecasts are exploited to train the meta-learner, which is effectively evaluated in the test set (red), again in a rolling origin fashion.

Table 3
Overall accuracy (RMSE and forecast skill index) per forecasting method and PV plant. The most accurate forecasting approach for each plant and accuracy measure is highlighted with boldface.

PV plant	Persistence	StackLSTM	BiLSTM	CNN-LSTM	ConvLSTM	EW	Meta
	RMSE						
Joao	4.555	2.393	2.569	2.435	2.415	2.377	**2.273**
Julia Moreira	2.764	1.587	1.645	1.719	1.634	1.601	**1.570**
Zarco	4.798	2.413	2.323	2.497	2.434	2.312	**2.298**
	Forecast skill index						
Joao	0	0.475	0.436	0.465	0.470	0.478	**0.500**
Julia Moreira	0	0.426	0.405	0.378	0.409	0.421	**0.432**
Zarco	0	0.497	0.516	0.479	0.493	0.518	**0.521**

method. Therefore, the natural benchmarks of our study are the four base models described in Section 3.

Nevertheless, as discussed in Section 1, ensembles of forecasts typically result in more accurate results than the forecasts they consist of. This is particularly true for simple combinations that build on the mean or the median of the base forecasts [27]. To evaluate whether the meta-learner adds value compared to such an approach, the equal weighted (EW) combination of the four base models is considered as an additional benchmark.

4.5. Results and discussion

Table 3 summarizes per PV plant the overall accuracy of the meta-learner, the four base forecasting models, and the EW benchmark in terms of RMSE and forecast skill index.

By focusing on the performance of the base forecasting models we observe that none of the forecasting approaches is dominant, meaning that different base models perform best depending on the PV plant being examined. On average, StackLSTM seems to provide more accurate results, followed by ConvLSTM and BiLSTM. However, the differences between the individual base models are neither consistent nor significant. This finding supports the need for optimally combining the forecasts of multiple models in the seek of more generalized and accurate forecasting solutions.

We also find that, according to RMSE, the meta-learner manages to provide more accurate results than the base forecasting models in all the examined PV plants, improving accuracy by up to 5% over the most accurate base model per plant. Similarly, depending on the PV plant, the meta-learner outperforms the EW benchmark by 0.6% to 4.4%. On the contrary, we observe that although EW is on average more accurate than the base models, in some cases the benchmark fails to outperform the most accurate individual model. Therefore, we conclude that the proposed meta-learning approach can effectively blend the forecasts of multiple models, outperforming standard yet competitive combinations approaches.

To conclude whether the meta-learner provides significantly better forecasts, we employ the multiple comparisons with the best (MCB) test [119]. The MCB test computes the average ranks of the forecasting methods according to RMSE across the complete set of hourly forecasts where the actual PV production is greater than zero and concludes whether or not these ranks are statistically different. Fig. 12 presents the results of the analysis for each PV plant separately. If the intervals of two methods do not overlap, this indicates a statistically different performance. Thus, methods that do not overlap with the gray interval of the figure are considered significantly worse than the best, and vice versa.

The results of the MCB test are particularly promising, suggesting that the meta-learner provides significantly more accurate forecasts than any of the benchmarks examined in two of the three PV plants. In the third park, “Julia Moreira”, the meta-learning approach is also ranked first, but its difference to the EW and StackLSTM approaches is statistically insignificant.

As a final step to our analysis, we investigate how forecasting accuracy varies depending on the realized capacity factor. This analysis can provide further insights under which circumstances the meta-learner is expected to add more value in term of forecasting accuracy improvements. Fig. 13 summarizes the accuracy of the meta-learner and the benchmarks according to RMSE for different realized capacity

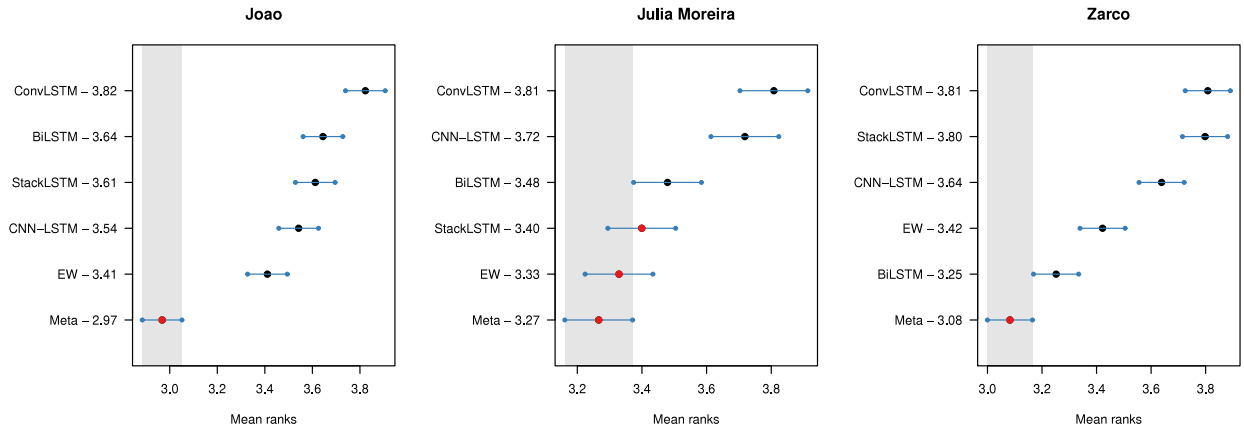


Fig. 12. Nemenyi test results at a 5% significance level for the RMSE. The tests involve records where the actual PV production is greater than zero.

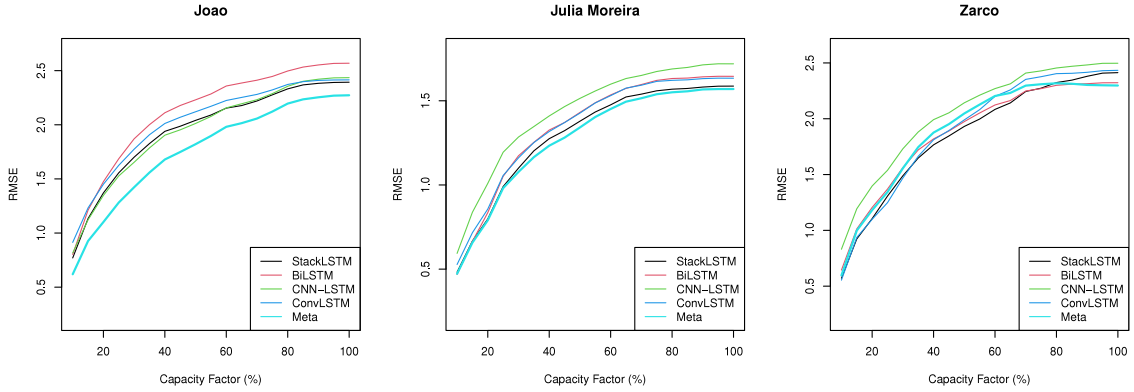


Fig. 13. Forecasting accuracy (RMSE) for different realized capacity factors. RMSE of capacity factor x % corresponds to the average forecast errors of capacity factors up to x %.

factors. As seen, in most of the cases the accuracy of the meta-learner is either better or on par with that of the benchmarks, with most of the improvements coming when the capacity factor is relatively higher. This observation supports our initial finding that the meta-learner is more likely to improve forecasting accuracy at peak hours.

5. Conclusions

This article presents an integrated method for short-term PV production forecasting. The developed method introduces four LSTM models of different deep architectures that can accurately forecast PV production in the following hour. The forecasts of these models serve as the base of a meta-learning model, tasked to optimally combine these forecasts given additional external features as input, such as weather and seasonal variables. Effectively, the meta-learner blends the base forecasts by exploiting the strengths of each base model, improving the overall accuracy and generalization of the proposed solution. Also, the presented method is NWP independent in the sense that the utilized weather features refer to previous time steps, thus being more reliable and fast to develop and implement.

The proposed methodology has been evaluated using data from three rooftop PV plants located in Lisbon, Portugal. Our results confirm that different base models perform best at different plants, although StackLSTM seems to result in more accurate forecasts on average, followed by ConvLSTM and BiLSTM. Moreover, our findings indicate that meta-learning can significantly improve average forecasting accuracy by up to 5% over the most accurate base model per PV plan, being at the same time up to 4.4% more accurate than an equal weighted forecast combination. Also, a meta analysis on the results suggests that meta-learning contributes most at peak production hours, i.e. when making accurate forecasts is most challenging and useful for decision makers. In such settings, the improvements over the worst performing base model can range from 10% up to 20%.

Future research could focus on examining how meta-learning can benefit multi-step PV power forecasting models. It is commonly agreed that the performance of NWP independent models significantly decays over a few hours, making them less reliable than NWP dependent models for e.g. day-ahead forecasting. Either way, meta-learning could still be used to enhance the performance of the base models used for making such forecasts. Moreover, future research could investigate other types of models as meta-learners and compare their performance to the proposed SVR method. Although the use of the SVR model is sufficiently justified, considering different features and base forecasts as meta-learner's inputs could potentially influence the selection of the meta-learner. More focus could also be put on the selection and development of the base models. DL is a powerful tool in the hands of forecasters and data scientists but its potential has not been fully exploited yet. As a result, further improving the performance of the base model through different DL architectures (e.g. transformers) and better feature selection and data preprocessing techniques could be part of the future work done in the field while it would be interesting to investigate the effect of the meta-learner in probabilistic forecasting settings instead of deterministic ones.

Finally, to enable industrial-scale implementation, our methodology offers scalability and adaptability, making it well-suited for deployment in industrial scale. Considerations for successful implementation include assessing computational requirements, integrating real-time data streams, and fostering collaborations with industry stakeholders. Validation and calibration on diverse industrial-scale PV plants can enhance reliability and generalizability. By addressing these aspects and closely collaborating with industry partners, our methodology has the potential to enhance short-term PV production forecasting, benefiting

decision-making, resource optimization, and operational efficiency in the renewable energy sector.

CRediT authorship contribution statement

Elissaios Sarmas: Conceptualization, Formal analysis, Software, Methodology, Writing – original draft. **Evangelos Spiliotis:** Data curation, Methodology, Visualization, Writing – original draft. **Efstathios Stamatopoulos:** Investigation, Formal analysis, Validation, Writing – original draft. **Vangelis Marinakis:** Conceptualization, Resources, Project administration, Writing – original draft. **Haris Doukas:** Investigation, Funding acquisition, Supervision, Writing – review & editing.

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.

We confirm that we have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing we confirm that we have followed the regulations of our institutions concerning intellectual property.

Acknowledgments

The work presented is based on research conducted within the framework of the project "Modular Big Data Applications for Holistic Energy Services in Buildings (MATRYCS)", of the European Union's Horizon 2020 research and innovation programme under grant agreement no. 1010000158 (https://matrycs.eu/) and of the Horizon 2020 European Commission project BD4NRG under grant agreement no. 872613 (https://www.bd4nrg.eu/). The authors wish to thank the Coopérnico team, whose contribution, helpful remarks and fruitful observations were invaluable for the development of this work. The content of the paper is the sole responsibility of its authors and does not necessary reflect the views of the EC.

References

[1] A. Olabi, M.A. Abdelkareem, Renewable energy and climate change, Renew. Sustain. Energy Rev. 158 (2022) 112111.
[2] J. Lee, M.M. Shepley, Benefits of solar photovoltaic systems for low-income families in social housing of Korea: Renewable energy applications as solutions to energy poverty, J. Build. Eng. 28 (2020) 101016.
[3] S. Jain, T. Sharma, A.K. Gupta, End-of-life management of solar PV waste in India: Situation analysis and proposed policy framework, Renew. Sustain. Energy Rev. 153 (2022) 111774.
[4] K. Reindl, J. Palm, Installing PV: Barriers and enablers experienced by non-residential property owners, Renew. Sustain. Energy Rev. 141 (2021) 110829.
[5] C. Wan, J. Zhao, Y. Song, Z. Xu, J. Lin, Z. Hu, Photovoltaic and solar power forecasting for smart grid energy management, CSEE J. Power Energy Syst. 1 (4) (2015) 38–46.
[6] J. Antonanzas, N. Osorio, R. Escobar, R. Urraca, F.J. Martinez-de Pison, F. Antonanzas-Torres, Review of photovoltaic power forecasting, Sol. Energy 136 (2016) 78–111.
[7] I. Maity, S. Rao, Simulation and pricing mechanism analysis of a solar-powered electrical microgrid, IEEE Syst. J. 4 (3) (2010) 275–284.
[8] M. Gürtler, T. Paulsen, The effect of wind and solar power forecasts on day-ahead and intraday electricity prices in Germany, Energy Econ. 75 (2018) 150–162.
[9] R. Ahmed, V. Sreeram, Y. Mishra, M. Arif, A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization, Renew. Sustain. Energy Rev. 124 (2020) 109792.
[10] L. Wen, K. Zhou, S. Yang, X. Lu, Optimal load dispatch of community microgrid with deep learning based solar power and load forecasting, Energy 171 (2019) 1053–1065.
[11] A. Tascikaraoglu, O. Erdinc, M. Uzunoglu, A. Karakas, An adaptive load dispatching and forecasting strategy for a virtual power plant including renewable energy conversion units, Appl. Energy 119 (2014) 445–453.
[12] J. Romaní, M. Belusko, A. Alemu, L.F. Cabeza, A. de Gracia, F. Bruno, Control concepts of a radiant wall working as thermal energy storage for peak load shifting of a heat pump coupled to a PV array, Renew. Energy 118 (2018) 489–501.
[13] R. Tassenoy, K. Couvreur, W. Beyne, M. De Paepe, S. Lecompte, Techno-economic assessment of Carnot batteries for load-shifting of solar PV production of an office building, Renew. Energy 199 (2022) 1133–1144.
[14] Y. Wu, Z. Liu, B. Li, J. Liu, L. Zhang, Energy management strategy and optimal battery capacity for flexible PV-battery system under time-of-use tariff, Renew. Energy 200 (2022) 558–570.
[15] B.H. Vu, I.-Y. Chung, Optimal generation scheduling and operating reserve management for PV generation using RNN-based forecasting models for stand-alone microgrids, Renew. Energy 195 (2022) 1137–1154.
[16] M. De Benedetti, F. Leonardi, F. Messina, C. Santoro, A. Vasilakos, Anomaly detection and predictive maintenance for photovoltaic systems, Neurocomputing 310 (2018) 59–68.
[17] A. Mellit, A.M. Pavan, A 24-h forecast of solar irradiance using artificial neural network: Application for performance prediction of a grid-connected PV plant at Trieste, Italy, Sol. Energy 84 (5) (2010) 807–821.
[18] A. Mellit, A.M. Pavan, V. Lughi, Deep learning neural networks for short-term photovoltaic power forecasting, Renew. Energy 172 (2021) 276–288.
[19] P. Bacher, H. Madsen, H.A. Nielsen, Online short-term solar power forecasting, Sol. Energy 83 (10) (2009) 1772–1783.
[20] P. Mathiesen, J. Kleissl, Evaluation of numerical weather prediction for intra-day solar forecasting in the continental United States, Sol. Energy 85 (5) (2011) 967–977.
[21] A. Mellit, A.M. Pavan, V. Lughi, Short-term forecasting of power production in a large-scale photovoltaic plant, Sol. Energy 105 (2014) 401–413.
[22] D. Chaturvedi, I. Isha, Solar power forecasting: A review, Int. J. Comput. Appl. 145 (6) (2016) 28–50.
[23] J.Á. González Ordiano, S. Waczowicz, M. Reischl, R. Mikut, V. Hagenmeyer, Photovoltaic power forecasting using simple data-driven models without weather data, Comput. Sci. Res. Dev. 32 (1) (2017) 237–246.
[24] F. Petropoulos, D. Apiletti, V. Assimakopoulos, M.Z. Babai, D.K. Barrow, S. Ben Taieb, C. Bergmeir, R.J. Bessa, J. Bijak, J.E. Boylan, J. Browell, C. Carnevale, J.L. Castle, P. Cirillo, M.P. Clements, C. Cordeiro, F.L. Cyrino Oliveira, S. De Baets, A. Dokumentov, J. Ellison, P. Fiszeder, P.H. Franses, D.T. Frazier, M. Gilliland, M.S. Gönül, P. Goodwin, L. Grossi, Y. Grushka-Cockayne, M. Guidolin, M. Guidolin, U. Gunter, X. Guo, R. Guseo, N. Harvey, D.F. Hendry, R. Hollyman, T. Januschowski, J. Jeon, V.R.R. Jose, Y. Kang, A.B. Koehler, S. Kolassa, N. Kourentzes, S. Leva, F. Li, K. Litsiou, S. Makridakis, G.M. Martin, A.B. Martinez, S. Meeran, T. Modis, K. Nikolopoulos, D. Önkal, A. Paccagnini, A. Panagiotelis, I. Panapakidis, J.M. Pavía, M. Pedio, D.J. Pedregal, P. Pinson, P. Ramos, D.E. Rapach, J.J. Reade, B. Rostami-Tabar, M. Rubaszek, G. Sermpinis, H.L. Shang, E. Spiliotis, A.A. Syntetos, P.D. Talagala, T.S. Talagala, L. Tashman, D. Thomakos, T. Thorarinsdottir, E. Todini, J.R. Trapero Arenas, X. Wang, R.L. Winkler, A. Yusupova, F. Ziel, Forecasting: theory and practice, Int. J. Forecast. 38 (3) (2022) 705–871.
[25] S. Sobri, S. Koohi-Kamali, N.A. Rahim, Solar photovoltaic generation forecasting methods: A review, Energy Convers. Manage. 156 (2018) 459–497.
[26] P. Montero-Manso, G. Athanasopoulos, R.J. Hyndman, T.S. Talagala, FFORMA: Feature-based forecast model averaging, Int. J. Forecast. 36 (1) (2020) 86–92.
[27] S. Makridakis, E. Spiliotis, V. Assimakopoulos, The M4 Competition: 100,000 time series and 61 forecasting methods, Int. J. Forecast. 36 (1) (2020) 54–74.
[28] U.K. Das, K.S. Tey, M. Seyedmahmoudian, S. Mekhilef, M.Y.I. Idris, W. Van Deventer, B. Horan, A. Stojcevski, Forecasting of photovoltaic power generation and model optimization: A review, Renew. Sustain. Energy Rev. 81 (2018) 912–928.
[29] P. Gupta, R. Singh, PV power forecasting based on data-driven models: a review, Int. J. Sustain. Eng. 14 (6) (2021) 1733–1755.
[30] S. Aslam, H. Herodotou, S.M. Mohsin, N. Javaid, N. Ashraf, S. Aslam, A survey on deep learning methods for power load and renewable energy forecasting in smart microgrids, Renew. Sustain. Energy Rev. 144 (2021) 110992.
[31] E. Spiliotis, N.Z. Legaki, V. Assimakopoulos, H. Doukas, M.S. El Moursi, Tracking the performance of photovoltaic systems: a tool for minimising the risk of malfunctions and deterioration, IET Renew. Power Gener. 12 (7) (2018) 815–822.
[32] H. Eom, Y. Son, S. Choi, Feature-selective ensemble learning-based long-term regional PV generation forecasting, IEEE Access 8 (2020) 54620–54630.
[33] M.J. Mayer, G. Gróf, Extensive comparison of physical models for photovoltaic power forecasting, Appl. Energy 283 (2021) 116239.
[34] J. Zhao, Z.-H. Guo, Z.-Y. Su, Z.-Y. Zhao, X. Xiao, F. Liu, An improved multi-step forecasting model based on WRF ensembles and creative fuzzy systems for wind speed, Appl. Energy 162 (2016) 808–826.
[35] A. Zameer, J. Arshad, A. Khan, M.A.Z. Raja, Intelligent and robust prediction of short term wind power using genetic programming based ensemble of neural networks, Energy Convers. Manage. 134 (2017) 361–372.
[36] M. Lei, L. Shiyan, J. Chuanwen, L. Hongling, Z. Yan, A review on the forecasting of wind speed and generated power, Renew. Sustain. Energy Rev. 13 (4) (2009) 915–920.

[37] H. Sharadga, S. Hajimirza, R.S. Balog, Time series forecasting of solar power generation for large-scale photovoltaic plants, Renew. Energy 150 (2020) 797–807.

[38] S.A. Haider, M. Sajid, H. Sajid, E. Uddin, Y. Ayaz, Deep learning and statistical methods for short-and long-term solar irradiance forecasting for Islamabad, Renew. Energy 198 (2022) 51–60.

[39] M.N. Akhter, S. Mekhilef, H. Mokhlis, N. Mohamed Shah, Review on forecasting of photovoltaic power generation based on machine learning and metaheuristic techniques, IET Renew. Power Gener. 13 (7) (2019) 1009–1023.

[40] M.J. Mayer, Benefits of physical and machine learning hybridization for photovoltaic power forecasting, Renew. Sustain. Energy Rev. 168 (2022) 112772.

[41] E. Ogliari, A. Dolara, G. Manzolini, S. Leva, Physical and hybrid methods comparison for the day ahead PV output power forecast, Renew. Energy 113 (2017) 11–21.

[42] A. Dolara, F. Grimaccia, S. Leva, M. Mussetta, E. Ogliari, A physical hybrid artificial neural network for short term forecasting of PV plant power output, Energies 8 (2) (2015) 1138–1153.

[43] D. Markovics, M.J. Mayer, Comparison of machine learning methods for photovoltaic power forecasting based on numerical weather prediction, Renew. Sustain. Energy Rev. 161 (2022) 112364.

[44] J. Wang, P. Li, R. Ran, Y. Che, Y. Zhou, A short-term photovoltaic power prediction model based on the gradient boost decision tree, Appl. Sci. 8 (5) (2018) 689.

[45] L. Visser, T. AlSkaif, W. van Sark, Operational day-ahead solar power forecasting for aggregated PV systems with a varying spatial distribution, Renew. Energy 183 (2022) 267–282.

[46] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.

[47] X. Luo, D. Zhang, X. Zhu, Combining transfer learning and constrained long short-term memory for power generation forecasting of newly-constructed photovoltaic plants, Renew. Energy 185 (2022) 1062–1077.

[48] E. Sarmas, E. Spiliotis, V. Marinakis, G. Tzanes, J.K. Kaldellis, H. Doukas, ML-based energy management of water pumping systems for the application of peak shaving in small-scale islands, Sustainable Cities Soc. 82 (2022) 103873.

[49] W. Lee, K. Kim, J. Park, J. Kim, Y. Kim, Forecasting solar power using long-short term memory and convolutional neural networks, IEEE Access 6 (2018) 73068–73080.

[50] K. Wang, X. Qi, H. Liu, A comparison of day-ahead photovoltaic power forecasting models based on deep learning neural network, Appl. Energy 251 (2019) 113315.

[51] H. Song, N. Al Khafaf, A. Kamoona, S.S. Sajjadi, A.M. Amani, M. Jalili, X. Yu, P. McTaggart, Multitasking recurrent neural network for photovoltaic power generation prediction, Energy Rep. 9 (2023) 369–376.

[52] Y. Wang, W. Liao, Y. Chang, Gated recurrent unit network-based short-term photovoltaic forecasting, Energies 11 (8) (2018) 2163.

[53] M. Abdel-Basset, H. Hawash, R.K. Chakrabortty, M. Ryan, PV-Net: An innovative deep learning approach for efficient forecasting of short-term photovoltaic energy production, J. Clean. Prod. 303 (2021) 127037.

[54] A. Rai, A. Shrivastava, K.C. Jana, A robust auto encoder-gated recurrent unit (AE-GRU) based deep learning approach for short term solar power forecasting, Optik 252 (2022) 168515.

[55] D. Lee, K. Kim, PV power prediction in a peak zone using recurrent neural networks in the absence of future meteorological information, Renew. Energy 173 (2021) 1098–1110.

[56] H. Zang, L. Cheng, T. Ding, K.W. Cheung, Z. Wei, G. Sun, Day-ahead photovoltaic power forecasting approach based on deep convolutional neural networks and meta learning, Int. J. Electr. Power Energy Syst. 118 (2020) 105790.

[57] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[58] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708.

[59] R. Zhu, W. Guo, X. Gong, Short-term photovoltaic power output prediction based on k-fold cross-validation and an ensemble model, Energies 12 (7) (2019) 1220.

[60] L. Liu, M. Zhan, Y. Bai, A recursive ensemble model for forecasting the power output of photovoltaic systems, Sol. Energy 189 (2019) 291–298.

[61] A.A. Lateko, H.-T. Yang, C.-M. Huang, H. Aprillia, C.-Y. Hsu, J.-L. Zhong, N.H. Phuong, Stacking Ensemble method with the RNN meta-learner for short-term PV power forecasting, Energies 14 (16) (2021) 4733.

[62] W. Khan, S. Walker, W. Zeiler, Improved solar photovoltaic energy generation forecast using deep learning-based ensemble stacking approach, Energy 240 (2022) 122812.

[63] A. Du Plessis, J. Strauss, A. Rix, Short-term solar power forecasting: Investigating the ability of deep learning models to capture low-level utility-scale Photovoltaic system behaviour, Appl. Energy 285 (2021) 116395.

[64] C.-H. Liu, J.-C. Gu, M.-T. Yang, A simplified LSTM neural networks for one day-ahead solar power forecasting, IEEE Access 9 (2021) 17174–17195.

[65] N.Y. Jayalakshmi, R. Shankar, U. Subramaniam, I. Baranilingesan, A. Karthick, B. Stalin, R. Rahim, A. Ghosh, Novel multi-time scale deep learning algorithm for solar irradiance forecasting, Energies 14 (9) (2021) 2404.

[66] J. Wang, Z. Qian, J. Wang, Y. Pei, Hour-ahead photovoltaic power forecasting using an analog plus neural network ensemble method, Energies 13 (12) (2020) 3259.

[67] F. Wang, Z. Xuan, Z. Zhen, K. Li, T. Wang, M. Shi, A day-ahead PV power forecasting method based on LSTM-RNN model and time correlation modification under partial daily pattern prediction framework, Energy Convers. Manage. 212 (2020) 112766.

[68] P. Li, K. Zhou, X. Lu, S. Yang, A hybrid deep learning model for short-term PV power forecasting, Appl. Energy 259 (2020) 114216.

[69] M. Mishra, P.B. Dash, J. Nayak, B. Naik, S.K. Swain, Deep learning and wavelet transform integrated approach for short-term solar PV power prediction, Measurement 166 (2020) 108250.

[70] Z. Yang, M. Mourshed, K. Liu, X. Xu, S. Feng, A novel competitive swarm optimized RBF neural network model for short-term solar power generation forecasting, Neurocomputing 397 (2020) 415–421.

[71] W. VanDeventer, E. Jamei, G.S. Thirunavukkarasu, M. Seyedmahmoudian, T.K. Soon, B. Horan, S. Mekhilef, A. Stojcevski, Short-term PV power forecasting using hybrid GASVM technique, Renew. Energy 140 (2019) 367–379.

[72] 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.

[73] G. Li, H. Wang, S. Zhang, J. Xin, H. Liu, Recurrent neural networks based photovoltaic power forecasting approach, Energies 12 (13) (2019) 2538.

[74] C.-J. Huang, P.-H. Kuo, Multiple-input deep convolutional neural network model for short-term photovoltaic power forecasting, IEEE Access 7 (2019) 74822–74834.

[75] L. Liu, Y. Zhao, D. Chang, J. Xie, Z. Ma, Q. Sun, H. Yin, R. Wennersten, Prediction of short-term PV power output and uncertainty analysis, Appl. Energy 228 (2018) 700–711.

[76] F. Rodríguez, A. Fleetwood, A. Galarza, L. Fontán, Predicting solar energy generation through artificial neural networks using weather forecasts for microgrid control, Renew. Energy 126 (2018) 855–864.

[77] J.F. Torres, A. Troncoso, I. Koprinska, Z. Wang, F. Martínez-Álvarez, Deep learning for big data time series forecasting applied to solar power, in: The 13th International Conference on Soft Computing Models in Industrial and Environmental Applications, Springer, 2018, pp. 123–133.

[78] S. Leva, A. Dolara, F. Grimaccia, M. Mussetta, E. Ogliari, Analysis and validation of 24 hours ahead neural network forecasting of photovoltaic output power, Math. Comput. Simulation 131 (2017) 88–100.

[79] M. Rana, I. Koprinska, V.G. Agelidis, Univariate and multivariate methods for very short-term solar photovoltaic power forecasting, Energy Convers. Manage. 121 (2016) 380–390.

[80] V. Sharma, D. Yang, W. Walsh, T. Reindl, Short term solar irradiance forecasting using a mixed wavelet neural network, Renew. Energy 90 (2016) 481–492.

[81] F. Almonacid, P. Pérez-Higueras, E.F. Fernández, L. Hontoria, A methodology based on dynamic artificial neural network for short-term forecasting of the power output of a PV generator, Energy Convers. Manage. 85 (2014) 389–398.

[82] F. Chollet, et al., Keras, 2015, https://keras.io.

[83] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[84] A. Sagheer, M. Kotb, Time series forecasting of petroleum production using deep LSTM recurrent networks, Neurocomputing 323 (2019) 203–213.

[85] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, T. Darrell, Long-term recurrent convolutional networks for visual recognition and description, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 2625–2634.

[86] S. Hochreiter, The vanishing gradient problem during learning recurrent neural nets and problem solutions, Int. J. Uncertain. Fuzziness Knowl.-Based Syst. 6 (02) (1998) 107–116.

[87] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, J. Schmidhuber, A novel connectionist system for unconstrained handwriting recognition, IEEE Trans. Pattern Anal. Mach. Intell. 31 (5) (2008) 855–868.

[88] A. Graves, Generating sequences with recurrent neural networks, 2013, arXiv preprint arXiv:1308.0850.

[89] C. Olah, Understanding lstm networks, 2015.

[90] H. Pang, L. Wu, J. Liu, X. Liu, K. Liu, Physics-informed neural network approach for heat generation rate estimation of lithium-ion battery under various driving conditions, J. Energy Chem. 78 (2023) 1–12.

[91] S. Wang, Y. Fan, S. Jin, P. Takyi-Aninakwa, C. Fernandez, Improved anti-noise adaptive long short-term memory neural network modeling for the robust remaining useful life prediction of lithium-ion batteries, Reliab. Eng. Syst. Saf. 230 (2023) 108920.

[92] Y. Bengio, et al., Learning deep architectures for AI, Found. Trends® Mach. Learn. 2 (1) (2009) 1–127.

[93] M. Hermans, B. Schrauwen, Training and analysing deep recurrent neural networks, Adv. Neural Inf. Process. Syst. 26 (2013).
[94] R. Pascanu, C. Gulcehre, K. Cho, Y. Bengio, How to construct deep recurrent neural networks, 2013, arXiv preprint arXiv:1312.6026.
[95] A. Graves, J. Schmidhuber, Framewise phoneme classification with bidirectional LSTM and other neural network architectures, Neural Netw. 18 (5–6) (2005) 602–610.
[96] M. Schuster, K.K. Paliwal, Bidirectional recurrent neural networks, IEEE Trans. Signal Process. 45 (11) (1997) 2673–2681.
[97] Y. Yao, Z. Huang, Bi-directional LSTM recurrent neural network for Chinese word segmentation, in: International Conference on Neural Information Processing, Springer, 2016, pp. 345–353.
[98] D. Wang, E. Nyberg, A long short-term memory model for answer sentence selection in question answering, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), 2015, pp. 707–712.
[99] M. Sajjad, S. Kwon, et al., Clustering-based speech emotion recognition by incorporating learned features and deep BiLSTM, IEEE Access 8 (2020) 79861–79875.
[100] T. Chen, R. Xu, Y. He, X. Wang, Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN, Expert Syst. Appl. 72 (2017) 221–230.
[101] Z. Li, F. Liu, W. Yang, S. Peng, J. Zhou, A survey of convolutional neural networks: analysis, applications, and prospects, IEEE Trans. Neural Netw. Learn. Syst. (2021).
[102] I.E. Livieris, E. Pintelas, P. Pintelas, A CNN–LSTM model for gold price time-series forecasting, Neural Comput. Appl. 32 (23) (2020) 17351–17360.
[103] W. Rawat, Z. Wang, Deep convolutional neural networks for image classification: A comprehensive review, Neural Comput. 29 (9) (2017) 2352–2449.
[104] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, W.-c. Woo, Convolutional LSTM network: A machine learning approach for precipitation nowcasting, Adv. Neural Inf. Process. Syst. 28 (2015).
[105] Z. Cui, R. Ke, Z. Pu, Y. Wang, Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction, 2018, arXiv preprint arXiv:1801.02143.
[106] B. Lindemann, T. Müller, H. Vietz, N. Jazdi, M. Weyrich, A survey on long short-term memory networks for time series prediction, Procedia CIRP 99 (2021) 650–655.
[107] J. Vanschoren, Meta-learning, in: Automated Machine Learning, Springer, Cham, 2019, pp. 35–61.
[108] E. Sarmas, E. Spiliotis, V. Marinakis, T. Koutselis, H. Doukas, A meta-learning classification model for supporting decisions on energy efficiency investments, Energy Build. 258 (2022) 111836.
[109] L. Todorovski, S. Džeroski, Combining multiple models with meta decision trees, in: European Conference on Principles of Data Mining and Knowledge Discovery, Springer, 2000, pp. 54–64.
[110] D.H. Wolpert, Stacked generalization, Neural Netw. 5 (2) (1992) 241–259.
[111] T. Hospedales, A. Antoniou, P. Micaelli, A. Storkey, Meta-learning in neural networks: A survey, 2020, arXiv preprint arXiv:2004.05439.
[112] B. Schölkopf, A.J. Smola, Learning with Kernel: Support Vector Machines, Regularization, Optimization and Beyond, The MIT Press, 2001.
[113] E. Spiliotis, F. Petropoulos, N. Kourentzes, V. Assimakopoulos, Cross-temporal aggregation: Improving the forecast accuracy of hierarchical electricity consumption, Appl. Energy 261 (2020) 114339.
[114] L.J. Tashman, Out-of-sample tests of forecasting accuracy: an analysis and review, Int. J. Forecast. 16 (4) (2000) 437–450.
[115] D. Koutsandreas, E. Spiliotis, F. Petropoulos, V. Assimakopoulos, On the selection of forecasting accuracy measures, J. Oper. Res. Soc. 73 (5) (2022) 937–954.
[116] D. Yang, Making reference solar forecasts with climatology, persistence, and their optimal convex combination, Sol. Energy 193 (2019) 981–985.
[117] D. Yang, A guideline to solar forecasting research practice: Reproducible, operational, probabilistic or physically-based, ensemble, and skill (ROPES), J. Renew. Sustain. Energy 11 (2) (2019) 022701.
[118] E. Sarmas, N. Dimitropoulos, V. Marinakis, Z. Mylona, H. Doukas, Transfer learning strategies for solar power forecasting under data scarcity, Sci. Rep. 12 (1) (2022) 1–13.
[119] A.J. Koning, P.H. Franses, M. Hibon, H. Stekler, The M3 competition: Statistical tests of the results, Int. J. Forecast. 21 (3) (2005) 397–409.