Energy 225 (2021) 120240

Contents lists available at ScienceDirect

Energy

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

Deep learning based forecasting of photovoltaic power generation by incorporating domain knowledge

Xing Luo [a, b], Dongxiao Zhang [c, a, *], Xu Zhu [b, d]

[a] Intelligent Energy Lab, Peng Cheng Laboratory, Shenzhen, 518055, PR China
[b] School of Electronic and Information Engineering, Harbin Institute of Technology, Shenzhen, 518055, PR China
[c] School of Environmental Science and Engineering, Southern University of Science and Technology, Shenzhen, 518055, PR China
[d] Department of Electrical Engineering and Electronics, University of Liverpool, Liverpool, L69 3GJ, UK

ARTICLE INFO

Article history:
Received 30 October 2020
Received in revised form
24 February 2021
Accepted 25 February 2021
Available online 10 March 2021

Keywords:
Solar energy
Forecasting
Domain knowledge
Physics-constrained LSTM

ABSTRACT

Solar energy constitutes an effective supplement to traditional energy sources. However, photovoltaic power generation (PVPG) is strongly weather-dependent, and thus highly intermittent. High-precision forecasting of PVPG forms the basis of the production, transmission, and distribution of electricity, ensuring the stability and reliability of power systems. In this work, we propose a deep learning based framework for accurate PVPG forecasting. In particular, taking advantage of the long short-term memory (LSTM) network in solving sequential-data based regression problems, this paper considers the specific domain knowledge of PV and proposes a physics-constrained LSTM (PC-LSTM) to forecast the hourly day-ahead PVPG. It aims to overcome the shortcoming of recent machine learning algorithms that are applied based only on massive data, and thus easily producing unreasonable forecasts. Real-life PV datasets are adopted to evaluate the feasibility and effectiveness of the models. Sensitivity analysis is conducted for the selection of input feature variables based on a two-stage hybrid method. The results indicate that the proposed PC-LSTM model possesses stronger forecasting capability than the standard LSTM model. It is more robust against PVPG forecasting, and more suitable for PVPG forecasting with sparse data in practice. The PC-LSTM model also demonstrates superior performance with higher accuracy of PVPG forecasting compared to conventional machine learning and statistical methods.

1. Introduction

The development of the global economy continually increases electricity demand and creates a huge impact on the environment. The massive usage of fossil fuels causes severe problems in terms of greenhouse gas emissions, which further leads to global warming and climate change around the world [1,2]. In response, an increasing amount of electricity has been generated from renewable energy sources in the past decade, among which photovoltaic (PV) energy is a major source, with a high penetration rate in the energy market [3]. PV power is especially promising due to its potential and availability. The total amount of solar energy that the Earth receives from solar radiation at any one instant is approximately 1.8×10^{11} MW [4], which can fully meet the demand of human activity globally.

Accurate forecasting of PV power generation (PVPG) is extremely important, as it can constitute a decision-making tool in power system operations [5]. Indeed, it is beneficial for both power suppliers and power systems. Power suppliers need to obtain precise information about PVPG for setting up dedicated commercial offers, thus reducing production cost and increasing profit. Meanwhile, it can also mitigate the negative impact caused by PV power uncertainty, ensuring the stability and reliability of the power system [6]. However, the PV output of a system mainly depends on the intensity of solar irradiance and a variety of meteorological factors, which are usually both uncertain and ungovernable [7]. The power generation of a PV system varies dynamically with time due to the variability of meteorological factors. Therefore, an accurate and stable forecasting of PVPG is considerably difficult and remains challenging.

The forecasting of PVPG has been an essential issue throughout the development of the PV industry, and much research has been conducted on it. The forecasting techniques presented in the

* Corresponding author. School of Environmental Science and Engineering, Southern University of Science and Technology, Shenzhen, 518055, PR China.
E-mail address: zhangdx@sustech.edu.cn (D. Zhang).

https://doi.org/10.1016/j.energy.2021.120240

literature can be broadly classified into four major categories (shown in Fig. 1) as follows.

(1) Physical model, which is developed based on global irradiance on solar cells, and a set of mathematical equations that describe the physical state of the PV system [1]. For physical models, acceptable forecasting accuracy can be obtained when weather conditions remain stable, whereas accuracy cannot be guaranteed if weather conditions change markedly. In Refs. [8,9], several typical physical models were presented.
(2) Statistical model, which is usually specified as a mathematical relationship between one or more random variables and other non-random variables [10]. It constitutes a basis for both inferences and predictions. In PVPG forecasting, the auto-regressive moving average (ARMA) model [11] and its improved versions (e.g., ARIMA, ARMAX, and SARIMA) [12,13], the regression model [14–18], and the exponential smoothing model [19,20] are the most important representatives of statistical models.
(3) Machine learning (ML) model, which is developed based on the statistical model, has been applied to various fields of science and engineering, including PVPG forecasting, in recent years. It follows a process of preparing data, training an algorithm, and generating the ML model, and then making and refining predictions [21]. Frequently used ML models in PVPG forecasting include artificial neural networks (ANN) [22–24], support vector machine (SVM) [25], and extreme learning machine (ELM) [26].
(4) Hybrid model. In many cases, a single model is not capable of accurately forecasting PVPG due to the limitation of a standalone technique [1]. Therefore, the combination of two or more techniques, which is known as the hybrid model, can be an effective solution for those cases. The main motivation of hybrid models is to explore potential combinations of different topologies for improving accuracy, by taking advantage of each topology. Examples of hybrid models utilized in PVPG forecasting include ANN + analog ensemble (AnEn) models [27], wavelet transform (WT)+particle swarm optimization (PSO)+support vector machine (SVM) models [28], and generative adversarial networks (GAN)+ convolutional neural networks (CNN) models [29].

In recent years, the latest development of smart metering technologies has given rise to an enormous volume of data, which is beneficial for the evolution of ML models. ANN has become the most frequently used method in practice [24]. However, the traditional ANN is a fully-connected neural network (FCNN) that can only simulate point-to-point mapping between the input space and the output space. In terms of PVPG forecasting, this means that the power output at time t, is merely correlated to the input data at selected specific times. FCNNs do not take the correlations of time-series data into account. Consequently, a better choice is the recurrent neural network (RNN) [30], in which an internal self-looped structure exists, allowing the network to preserve previous information. In RNNs, the long short-term memory (LSTM) network is an outstanding representative. LSTM can not only handle the correlation of time-series as RNN, but also prevent gradient vanishing and gradient exploding, which are found with traditional RNNs. Recently, LSTM has been utilized in PVPG forecasting with promising results [5,31,32].

Despite numerous successes obtained from previous works, limitations remain in the research of PVPG forecasting. Recent ML models are applied based only on massive data and no domain knowledge or physical laws are involved in the construction of the model. However, for the standard LSTM, it may produce physically unreasonable predictions. This may occur in a deep neural network without incorporating domain knowledge and physical laws of a specific engineering problem [33]. In terms of PVPG forecasting, unreasonable predictions commonly occurred in training and testing processes include negative power generation, positive power generation at midnight, low solar radiation predicting high power generation, and high solar radiation predicting extremely low power generation [5,31,32], which may have negative impacts on the performance of a deep learning based forecasting model. Unfortunately, there is currently no such research on PVPG forecasting. Therefore, much room exists to improve the model in terms of predictability, reliability, and interpretability.

To overcome the above constraints, incorporating domain knowledge and physical laws of PV, a deep learning based framework, and the so-called physics-constrained LSTM (PC-LSTM) is proposed in this work to forecast hourly day-ahead PVPG based on selected feature data. The main contributions of our work are summarized as follows:

(1) Based on the proposed deep learning framework, physical constraints are extracted from domain knowledge of PV and then firstly integrated into the construction of the PC-LSTM model.
(2) A two-stage hybrid method combining filter and wrapper processes is proposed to theoretically select the highly correlated feature variables. A sensitivity analysis of PV data is conducted to ensure appropriate input for ML models.
(3) The proposed PC-LSTM model incorporating domain knowledge of PV possesses stronger forecasting capability in terms of robustness and sparse data based forecasting ability than the standard LSTM model.
(4) The proposed PC-LSTM model can achieve better performance with higher accuracy of PVPG forecasting compared to conventional ML and statistical methods.

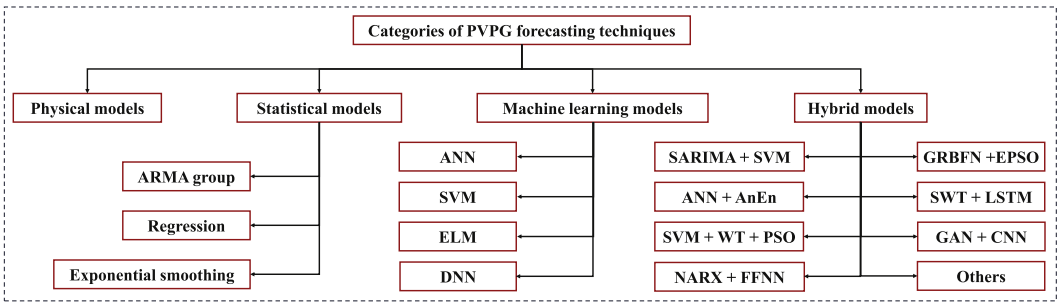

Fig. 1. Summary of PVPG forecasting techniques in the literature.

The remainder of this paper is organized as follows. In Section 2, a two-stage hybrid method is introduced to determine the appropriate feature variables. The standard LSTM and PC-LSTM models are proposed specifically in Section 3. In Section 4, the proposed approaches are evaluated based on real-life PV datasets. The forecasting performances of different models are also compared. Finally, the paper is concluded in Section 5.

2. Selection method of feature variables

Solar energy comes from the Sun in the form of solar irradiance through the photovoltaic effect. PV output predominantly relies on the intensity of solar irradiance. In addition, other meteorological variables, such as surface pressure, humidity, temperature, wind speed, and precipitation are also regarded as potential factors that may affect the output of a PV system. Therefore, among numerous features related to PV, it is necessary to identify which ones are important to the model and which ones are irrelevant, thus selecting an appropriate feature subset as the model input. In this work, we propose a hybrid method that comprises two major stages, namely, the filter stage and the wrapper stage, for feature variable selection, as illustrated in Fig. 2.

In the first stage, the filter method examines features based on intrinsic characteristics prior to the learning task. It selects a subset of features from the dataset according to different types of filter criteria, e.g., dependency, information, distance, and consistency [34]. Considering the characteristics of PV data, Pearson correlation coefficient (PCC) [35] is adopted to evaluate the relevance between input variables and the target variable. PCC is a measure of the linear correlation between variables X and Y in statistics. In time-series data, it describes to what extent the target variable Y correlates to the input variable X at different times. However, the correlation between the time-series data at time t and time $t-1$ of the variable is not calculated. In our case, the target variable Y represents the PVPG, whereas the input variables represent the meteorological variables related to PVPG. Therefore, PCC can be mathematically defined as follows:

$$\rho(X;Y)=\frac{\sum_{i=1}^{N}(x_i-\bar{x})\cdot(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i-\bar{x})^2}\cdot\sqrt{\sum_{i=1}^{N}(y_i-\bar{y})^2}} \tag{1}$$

where $\bar{x}=\sum_{i=1}^{N}x_i/N$ and analogously for $\bar{y}$. PCC has a value between $+1$ and -1, where $+1$ indicates the total positive linear correlation; 0 represents no linear correlation; and -1 denotes the total negative linear correlation between X and Y. It is important to note that PCC is one of the most widely used criteria to describe relevance between variables in practice. Other criteria such as Spearman coefficient, mutual information coefficient, and distance correlation coefficient also can be used for relevance evaluation between variables from different angles.

In the second stage, the wrapper method is used to evaluate each subset. Specifically, it wraps the feature selection around the learning algorithm, and determines the most appropriate feature subset by minimizing the forecasting error of a particular forecasting model. In this work, feature subsets are evaluated based on the standard LSTM model due to the advantages of LSTM in solving time-series forecasting problems. Therefore, the best training feature subset among all evaluated subsets can be finally determined.

The proposed hybrid method for feature selection attempts to inherit the advantages of both filter and wrapper methods by combining their complementary strengths. In the first stage, correlations between variables are assessed individually via filter criteria, and feature variables can be filtered accordingly by establishing proper thresholds. The filter method is univariate and independent of any learning algorithm. As a result, it is very efficient and computationally faster than the wrapper method, and able to process large datasets. However, interactions with learning algorithms and among features are ignored at this stage. Combined features may have net effects that are not necessarily reflected by the single feature evaluation, so that the wrapper method is required. The individual wrapper method always consumes extensive computational resources, since a learning algorithm is applied and a great number of feature subsets need to be evaluated. Nonetheless, by using correlation results obtained by the filter method as guidance for generating feature subsets, fewer subsets are generated and evaluated. Therefore, the proposed hybrid method can improve the efficiency of feature selection to a certain degree.

The proposed two-stage hybrid method provides an effective solution for the selection of feature variables and ensures appropriate input for ML models. The feature selection result for PV data is presented in Subsection 4.2. In the next section, PVPG forecasting models are illustrated in detail.

3. PV power generation forecasting models

In Section 3, the standard LSTM model is introduced first. Subsequently, the physics-constrained LSTM model is proposed.

3.1. LSTM model

LSTM is a special kind of RNN, which differs fundamentally from conventional ANNs. Both LSTM and RNN are sequence-based models with internal self-looped repeating networks (illustrated in Fig. 3), which can preserve previous information and establish temporal correlations between sequential data. The main difference between LSTM and RNN is the structure of repeating modules. In the basic RNN, the repeating module has a very simple structure (e.g., *Tanh* layer), while LSTM contains four interacting layers (three gate layers and a *Tanh* layer) within its repeating modules. The architecture of an LSTM block is demonstrated in Fig. 4.

In LSTM, the cell state is a core variable that can run straight

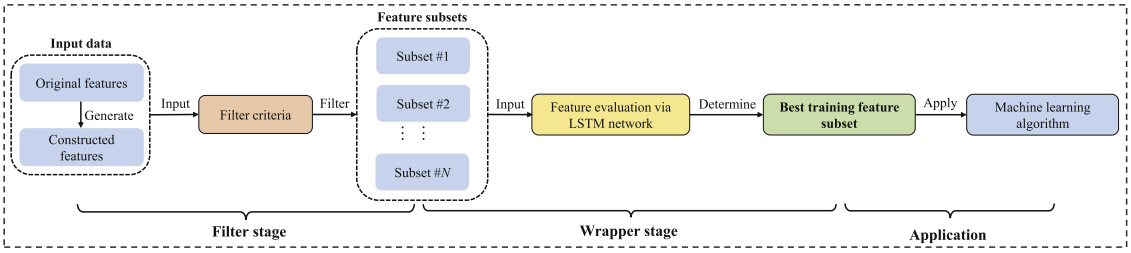

Fig. 2. Block diagram of feature selection based on the two-stage hybrid method.

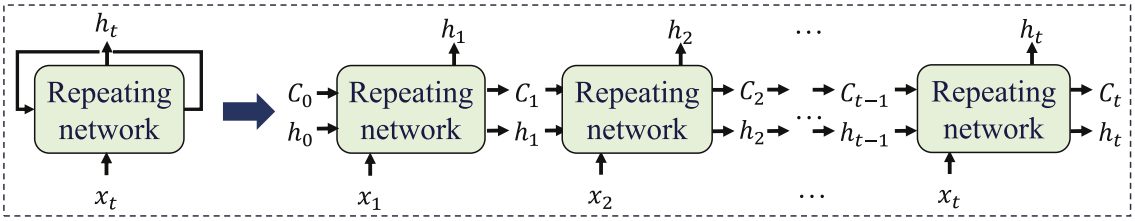

Fig. 3. Illustration of an unrolled repeating network of LSTM.

down the entire network, carrying information of previous steps. The LSTM model possesses the ability to remove or add information to the cell state, which is carefully regulated by the gate layer structure. Specifically, the first layer is called the "forget gate" layer, which determines what information from previous steps to forget. The output of the forget gate, f_t, is a number between zero and one. It can be mathematically expressed as:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{2}$$

The second layer of the LSTM block is called the "input gate" layer. It is used to determine what new information that will be stored in the cell state. It can be illustrated in Eq. (3).

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{3}$$

The third layer is the *Tanh* layer. The *Tanh* layer can be used to generate a vector of new candidate values that could be added to the state. It is defined in Eq. (4).

$$\tilde{C}_t = \varphi(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{4}$$

After the first three layers, the old cell state, C_{t-1}, has to be updated into the new cell state, C_t. The output of the forget gate, f_t, decides what to forget, and the result of the input gate, i_t, determines what to add to the new cell state, $\tilde{C}_t$. The updating process of C_t is formulated as follows:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{5}$$

Finally, the last interacting layer, which is termed the "output gate" layer, generates the final output according to the updated cell state. The process of generating an output of the LSTM block is illustrated in Eq. (6).

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) * \varphi(C_t) \tag{6}$$

where σ is the activation function, such as *Sigmoid*; and φ denotes the *Tanh* function. Let $\theta = \{W, b\}$ be the parameter vector of network, $W = [W_f, W_i, W_C, W_o]$ and $b = [b_f, b_i, b_C, b_o]$ are the weights and bias of each layer, respectively. The forward formulation in Eqs. (2)–(6) can be simply denoted as $Y = NN(X; \theta)$. Afterwards, the loss function of the LSTM model, $\mathscr{L}(\theta)_{\text{LSTM}}$, which is usually the mean square error (MSE) between the output and the ground truth data, can be illustrated as follows:

$$\mathscr{L}(\theta)_{\text{LSTM}} = \frac{1}{N}\sum_{i=1}^{N} |\text{NN}(x_i; \theta) - y_i|^2 \tag{7}$$

where N is the total number of labeled data. During the training process of LSTM, the parameter vector of network, θ, will be tuned continuously by minimizing the loss function via an optimization algorithm, such as the stochastic gradient descent (SGD).

3.2. Physics-constrained LSTM model

In this subsection, physical constraints are extracted from domain knowledge and physical laws of PV at first, and then

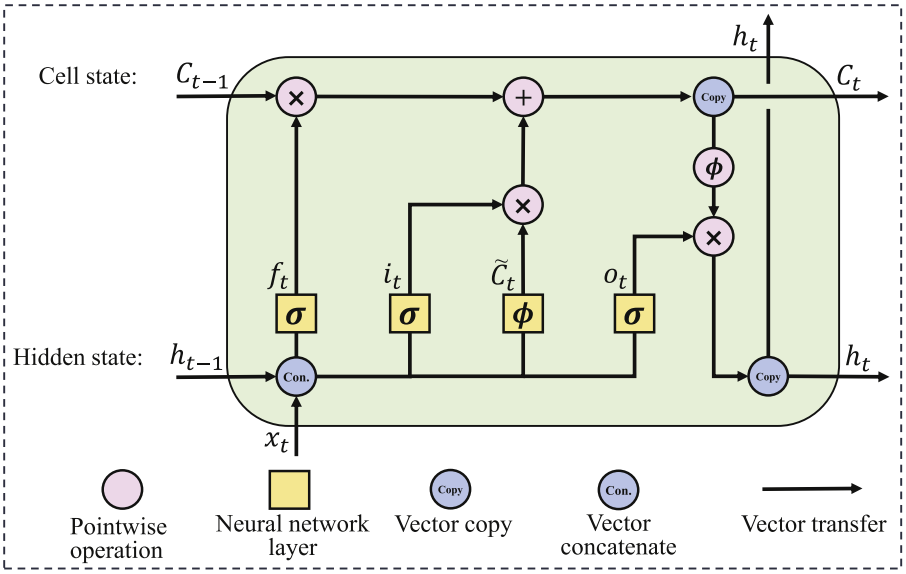

Fig. 4. Architecture of an LSTM block with four interacting layers.

integrated into the construction of the PC-LSTM model. It aims to overcome the shortcoming of recent machine learning algorithms that are applied based only on massive data without considering the impacts of physical regulations. The architecture of the PC-LSTM model is presented in Fig. 5.

Specifically, three types of constraints (indicated as Cons. #1, Cons. #2, and Cons. #3 in Fig. 5) are integrated into the construction of PC-LSTM. The first module is called the *Data Filtering Module*, which filters the input data into different periods of time according to a flag variable. It is designed based on world knowledge or general knowledge of PV [36] and aims to eliminate physically unreasonable forecasts, such as positive power generation at midnight, during training and testing processes, via filtering training data.

As discussed in Section 2, PV power production is significantly dependent on the solar radiation received by the PV panels near the land surface. Consequently, it is important to flag periods that have positive solar radiation at the surface. This is accomplished automatically by the data filtering module based on values of the time-series variable, termed hourly surface radiation (SR^*). In the training stage, only the data in flagged periods can be transferred into the model for further training and be used to forecast the PV output. On the other hand, in the forecasting stage, for periods when the surface radiation is expected to be zero, the resulting PV output will be calculated accordingly. Due to less data being adopted for model training, the efficiency of PC-LSTM can be improved to a certain degree.

The second constraint integrated into the PC-LSTM is called the *Clipping Module*, which is used to restrict the output of the model within a reasonable range in both training and testing processes. It is designed based on natural science knowledge of PV [36], and aims to eliminate physically unreasonable forecasts, such as negative power generation. According to the physical law, the value of PVPG should be physically greater than zero in practice, and therefore the model output should be positive. As a result, the output of PC-LSTM, $\widehat{y}_i$, should be subject to the constraint in Eq. (8).

$$\widehat{y}_i := \mathrm{ReLU}(\mathrm{NN}(x_i;\theta)) \tag{8}$$

where $\mathrm{ReLU}(\cdot)$ is the rectified linear unit function. The ReLU function returns zero when the input to the function is negative, and returns the original value of the input when it is positive.

Engineering controls in practice may also assist to guide the construction of PC-LSTM. As PV power is converted directly from surface radiation, the amount of PVPG for a certain period of time should theoretically fall into a certain range while the amount of solar radiation is determined. According to the photoelectric conversion relationship between SR^* and PVPG, as illustrated in Fig. 6, the PV power output should fall within certain bounds, which may be constructed with historical data or on theoretical grounds. From Fig. 6, the PVPG values vary significantly within the bounds. This is so because the PVPG is not only related to the surface radiation, but also reflects the physical information of PV panels, such as material, orientation, and geographic information. Moreover, the outliers, marked as black crosses in the figure, can be detected by using the K-means algorithm [37] based on a proper evaluation criterion. In this study, if the distance between the evaluated point and centroid of a cluster is over a threshold δ, where $\delta = \bar{x} + 3\sigma$, the point is regarded as an outlier.

Subsequently, the upper and lower bounds can be determined by bounding points (without outliers) among historical data. As a result, upper and lower bound functions are defined as bound controls to restrict the forecasts of PV output during the training process. In this study, the rational function, which can be denoted as an algebraic fraction with numerator and denominator both given by polynomial expressions, is utilized to derive the lower and upper bound functions as $f^{\mathrm{LB}}(x)$ and $f^{\mathrm{UB}}(x)$, respectively. The analytic forms of bound functions adopted in this work are illustrated specifically in the *Appendix*. It is noted that the rational function is not the only format that can be used for the fitting task. Fitting functions in other formats, such as polynomial, exponential, and power with appropriate coefficients, are also acceptable. Afterwards, the constraint in the form of bound control will be integrated into the construction of the PC-LSTM model.

Integration of the above constraint into the PC-LSTM model is by reconstructing the loss function via the *Loss Penalty Module*. Theoretically, when those bound controls are violated, there should be a penalty loss term or knowledge-based loss term $\mathrm{MSE}_{\mathrm{PLT}}$, reflected in the loss function [36]. Therefore, the loss function of the PC-LSTM model can be reformulated as:

$$\mathscr{L}(\theta)_{\mathrm{PC-LSTM}} = \mathrm{MSE}_{\mathrm{DATA}} + \mathrm{MSE}_{\mathrm{PLT}} \tag{9}$$

where

$$\mathrm{MSE}_{\mathrm{DATA}} = \frac{1}{N_{\mathrm{DATA}}} \sum_{i=1}^{N_{\mathrm{DATA}}} \left| \widehat{y}_i - y_i \right|^2 \tag{10}$$

$$\mathrm{MSE}_{\mathrm{PLT}} = \frac{1}{N_{\mathrm{PLT}}} \sum_{i=1}^{N_{\mathrm{PLT}}} \lambda_{\mathrm{PLT},\,i} \cdot \left| \widehat{y}_i - y_i^* \right|^2 \tag{11}$$

The total loss of PC-LSTM is reconstructed as the combination of data loss and penalty loss, as shown in Eq. (9). In Eq. (11), $\lambda_{\mathrm{PLT},\,i}$ is an additional hyper-parameter of PC-LSTM, which denotes the intensity of penalty on the loss function, and $y_i^* = \frac{1}{2}(f^{\mathrm{LB}}(x_i) + f^{\mathrm{UB}}(x_i))$.

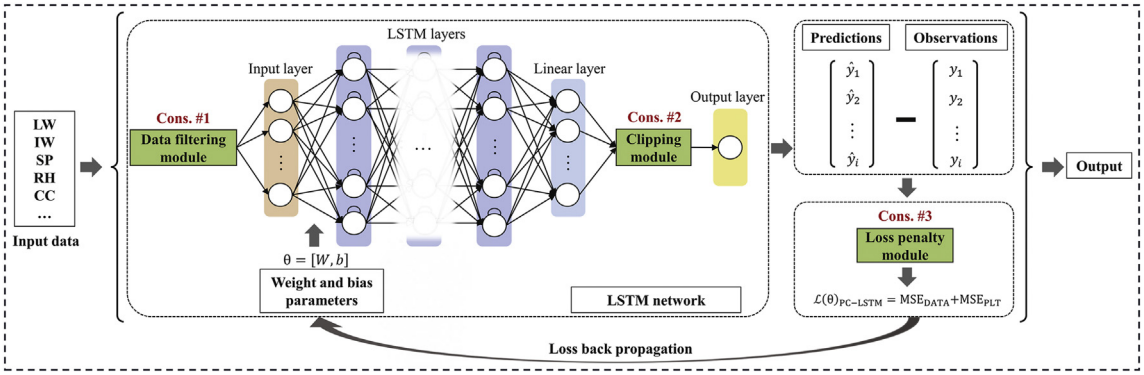

Fig. 5. Architecture of the PC-LSTM model with three physical constraint modules.

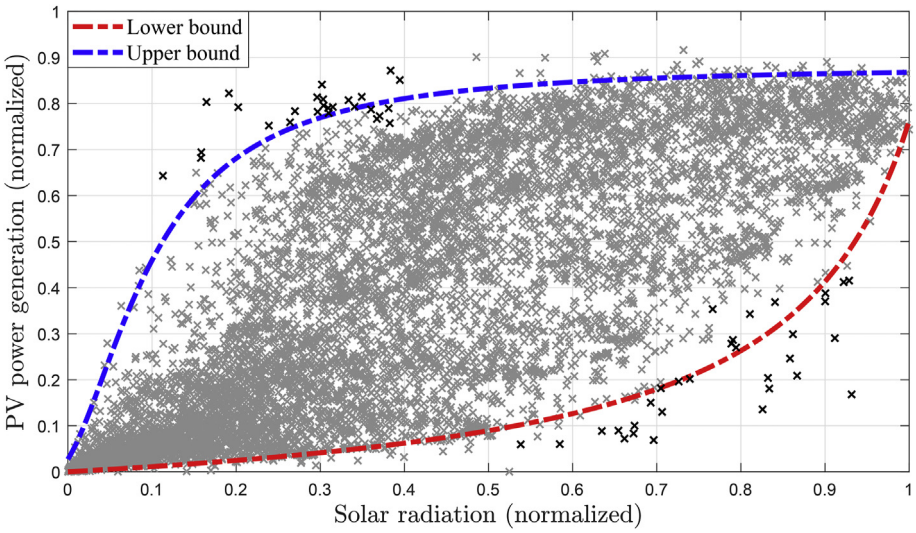

Fig. 6. Photoelectric conversion relationship between hourly solar radiation and PVPG of a typical PV plant.

In the training process, when an output after the clipping module in the neural network, $\widehat{y}_i$ (explained in Eq. (8)), satisfies the bound controls, $\lambda_{\mathrm{PLT},\,i} = 0$ and $\mathrm{MSE}_{\mathrm{PLT}} = 0$, this means that the model is trained as it used to be. In contrast, when the bound controls are violated, $\lambda_{\mathrm{PLT},\,i} > 0$ and $\mathrm{MSE}_{\mathrm{PLT}} > 0$, this indicates that additional penalties will be added to the loss function of PC-LSTM. Meanwhile, the parameter vector, $\theta = \{W, b\}$, will be tuned accordingly, and the model will be updated simultaneously. Therefore, $\lambda_{\mathrm{PLT},\,i}$ can be defined as follows:

$$\begin{cases} \lambda_{\mathrm{PLT},\,i} = 0,\ \widehat{y}_i \in \left[f^{\mathrm{LB}}(x_i),\ f^{\mathrm{UB}}(x_i)\right] \\ \lambda_{\mathrm{PLT},\,i} > 0,\ \widehat{y}_i \notin \left[f^{\mathrm{LB}}(x_i),\ f^{\mathrm{UB}}(x_i)\right] \end{cases} \tag{12}$$

The proposed constraint modules coordinate with each other to ensure that all outputs of the PC-LSTM model are reasonable during training and testing processes. The PC-LSTM model is trained with available observations, while being simultaneously guided by the above domain knowledge and engineering/physical constraints. In the next section, case studies are carried out to verify the feasibility and effectiveness of the proposed methods based on real-life PV datasets.

4. Case study

In Section 4, several case studies are carried out to evaluate the feasibility and effectiveness of the proposed methods. Real-life datasets from two typical PV plants are adopted.

4.1. Data pre-processing

4.1.1. Data description

In this study, installation parameters of two individual PV plants, which are located in different areas in Australia, are described specifically in Table 1 [38]. The datasets from two PV plants are adopted to simultaneously evaluate the proposed approaches to ensure convincible results. The feature dataset contains weather forecasts for 12 independent weather variables, which are obtained from the European Center for Medium-range Weather Forecasts (ECMWF) [38]. Because the weather data are of hourly resolution, the PVPG forecasting is also hourly. Different resolutions of forecasting can be made with the PC-LSTM model when corresponding weather data are available. Additionally, the forecasting horizon of PVPG depends entirely on the horizon of weather forecasts that can be obtained in advance. In this work, it is assumed that one-day ahead accurate weather forecasts are available. Therefore, the PVPG forecasting horizon for all compared models can be regarded as day-ahead. The weather variables utilized in this work are illustrated specifically in Table 2.

Table 1
Installation parameters of adopted PV plants.

Parameters	PV index	
	Plant #1	Plant #2
Altitude	595 m	951 m
Panel type	Solarfun SF160-24-1M195	Suntech STP200-18/ud
Panel number	8	20
Nominal power	1560 W	4000 W
Orientation	38°, Clockwise from North	31°, Clockwise from North
Panel tilt	36°	21°

4.1.2. Feature construction

Based on the illustration of Table 2, four feature variables, namely, surface solar radiation downwards (SR), surface thermal radiation downwards (TR), top-net solar radiation (TS), and total precipitation (TP), are recorded as daily accumulated values; whereas, the PVPG data are collected hourly. Consequently, it is necessary to convert these variables to the hourly accumulated field as SR^*, TR^*, TS^*, and TP^* by the first order difference. Using SR^* as an example, since $\mathrm{SR}_t = \sum_{t=1}^{t} \mathrm{SR}_t^*$, the process of converting SR to SR^* can be expressed as: $\mathrm{SR}_t^* = \mathrm{SR}_t - \mathrm{SR}_{t-1}$, where $t = 1, 2, \ldots, 24$. The units of SR^*, TR^*, and TS^* also have to be converted to $\mathrm{W \cdot m^{-2}}$. In addition, a binary feature variable, i.e., $\mathrm{DN} = [0, 1]$, is constructed to indicate day and night time patterns of time-series data ("0" = night-time pattern and "1" = day-time pattern). This is accomplished by a visual inspection of the SR time series. As a result, five additional feature variables are constructed, and 17 feature variables in total can be potentially utilized as model input.

4.1.3. Data normalization

The goal of data normalization is to process the data entries in

Table 2
Illustration of available feature variables within the initial dataset.

ID	Feature name	Unit	Comment(s)
LW	Total column liquid water	$kg \cdot m^{-2}$	Vertical integral of cloud liquid water content.
IW	Total column ice water	$kg \cdot m^{-2}$	Vertical integral of cloud ice water content.
SP	Surface pressure	Pa	–
RH	Relative humidity at 1000 mbar	%	RH is defined with respect to saturation of the mix phase.
CC	Total cloud cover	0–1	CC derived from model using the model's overlap assumption.
10U	10-m U wind component	$m \cdot s^{-1}$	–
10V	10-m V wind component	$m \cdot s^{-1}$	–
2T	2-m temperature	K	–
SR	Surface solar radiation downwards	$J \cdot m^{-2}$	Accumulated value.
TR	Surface thermal radiation downwards	$J \cdot m^{-2}$	Accumulated value.
TS	Top-net solar radiation	$J \cdot m^{-2}$	Net solar radiation at the atmosphere. Accumulated value.
TP	Total precipitation	m	Convective + stratiform precipitation. Accumulated value.

various ranges into a consistent range, thus minimizing regression error while maintaining correlation among the dataset. In this study, the min-max normalization technique, which restricts the data within the range between zero and one, is adopted. The formula of the min-max normalization is illustrated as:

$$x_{norm} = \frac{x_{orig} - x_{min}}{x_{max} - x_{min}} \tag{13}$$

where x_{norm} and x_{orig} are the normalized data and original data, respectively; and symbols x_{min} and x_{max} denote the minimum and maximum values of a specific feature within the dataset, respectively. All feature data should be normalized prior to transmitting to the model. It is noted that the data of the target variable have already been processed within the dataset.

4.1.4. Data splitting for cross-validation

Cross-validation is an effective way to ensure that forecasting models are robust. In cross-validation, the initial dataset is divided into two sets: the training dataset and the testing dataset. The bulk of data is used for training the model, while the testing dataset is used to evaluate the model to prevent over-fitting. In this work, the historical data of 730 days (from 01 to 04–2012 to 31-03-2014, 2 years) are set as the training dataset, whereas the data of 91 days (from 01 to 04–2014 to 30-06-2014, 3 months) are set as the testing dataset. The method of cross-validation ensures that any improvements achieved by the proposed model based on the training data will also lead to an improvement for the testing data, thus enhancing the robustness of the model. The PVPG bounds shown in Fig. 6 are constructed with the training data, and are used to guide the training process via Eqs. 9–11.

4.1.5. Evaluation metrics

In order to evaluate the forecasting performances, several evaluation metrics, including MAE, MSE, and R^2 score, are proposed. The mathematical formulations of MAE, MSE, and R^2 score can be expressed in Eqs. 14–16, respectively.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|\widehat{y}_i - y_i\right| \tag{14}$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(\widehat{y}_i - y_i\right)^2 \tag{15}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}\left(\widehat{y}_i - y_i\right)^2}{\sum_{i=1}^{N}\left(\overline{y} - y_i\right)^2} \tag{16}$$

where $\widehat{y}_i$, y_i, and $\overline{y}$ represent the predicted value, observed value, and the mean of observed values, respectively.

4.2. Feature selection of PV data

One necessary and significant task prior to PVPG forecasting is to analyze feature variables and select an appropriate feature subset as the model input. As discussed in Section 2, the two-stage hybrid method is introduced to determine the appropriate feature variables. In the first stage, the PCC criterion is utilized to evaluate the relevance between input variables and the target variable. The correlation results illustrated in Fig. 7 show that the feature variables SR^* and TS^*, both related to solar radiation, have strong correlations with PVPG, which is in accordance with the previous discussion.

In the second stage, the initial dataset is then divided into different subsets according to the intensity of correlation. As a result, six feature subsets are formed, as shown in Table 3. Afterwards, feature subsets are evaluated individually multiple times ($N_{test} = 10$) based on the standard LSTM model, so that the best subset among tests can be determined accordingly.

The box-plot results of subset evaluation for two individual PV plants are presented in Fig. 8. The results in Fig. 8 (a) and (b) are similar and show that Subset #1 performs the best among all evaluated subsets with the lowest MSE, whereas Subset #4 performs the worst among tests. It indicates that subsets consisting of more feature variables may demonstrate a better result with lower MSE compared to subsets consisting of fewer feature variables, in general. It is also interesting to observe that the LSTM model can obtain acceptable forecasting accuracy by using Subset #5, which only consists of two feature variables related to solar radiation, i.e., SR^* and TS^*. Furthermore, when comparing the results of Subset #1 and Subset #6, it is shown that properly constructed new variables are significant in feature selection, and a certain degree of improvement can be obtained. Therefore, the Subset #1 is adopted as the input to models in the following case studies.

4.3. Forecasting capability of PC-LSTM

As the PC-LSTM model is developed based on the standard LSTM model by incorporating some domain knowledge of PV, it is necessary to compare the forecasting performance of the PC-LSTM model versus the standard LSTM model. Consequently, in this

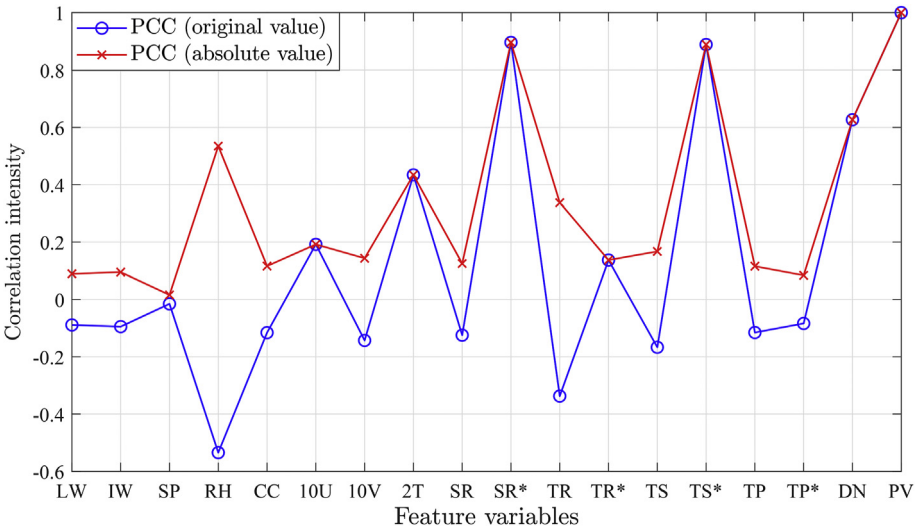

Fig. 7. Evaluation of correlations between the target variable and feature variables.

Table 3
Illustration of formed feature subsets.

Subset ID	Features (number)	PCC threshold
Subset #1	All of feature variables (17)	–
Subset #2	Subset #1 excludes LW, IW, SP, and TP* (13)	|PCC|>0.1
Subset #3	Subset #2 excludes CC, SR, TR*, TS, and TP (8)	|PCC|>0.15
Subset #4	Subset #3 excludes 10U and 10V (6)	|PCC|>0.3
Subset #5	SR* and TS* (2)	|PCC|>0.7
Subset #6	Original feature variables (12)	–

subsection, the robustness and sparse data based forecasting ability of models are compared between PC-LSTM and LSTM, and sensitivity analyses are proposed. In all comparisons, each experiment is conducted for 10 times to reduce the impact of randomness. The MSE is used as the main criterion in such experiments.

4.3.1. Forecasting capability comparison between PC-LSTM and standard LSTM - robustness

In deep learning, the setting of hyper-parameters of the model may have significant impacts on model performance. To evaluate the robustness of the proposed PC-LSTM model, eight independent cases with different settings of hyper-parameters are utilized. In these cases, the hyper-parameters related to model structure are fixed, whereas other hyper-parameters (i.e., learning rate and batch size) are adjusted. For each case, the models are evaluated based on the same settings of hyper-parameters and the same training data from two individual PV plants (explained in Subsection 4.1) to ensure convincible results.

The results of the robustness comparison between LSTM and PC-LSTM are illustrated specifically in Fig. 9. The blue boxes and deep grey boxes represent the MSE loss of PC-LSTM and LSTM, respectively. The red crosses denote the averaged MSE for 10 evaluations, whereas the red pluses are outliers produced by

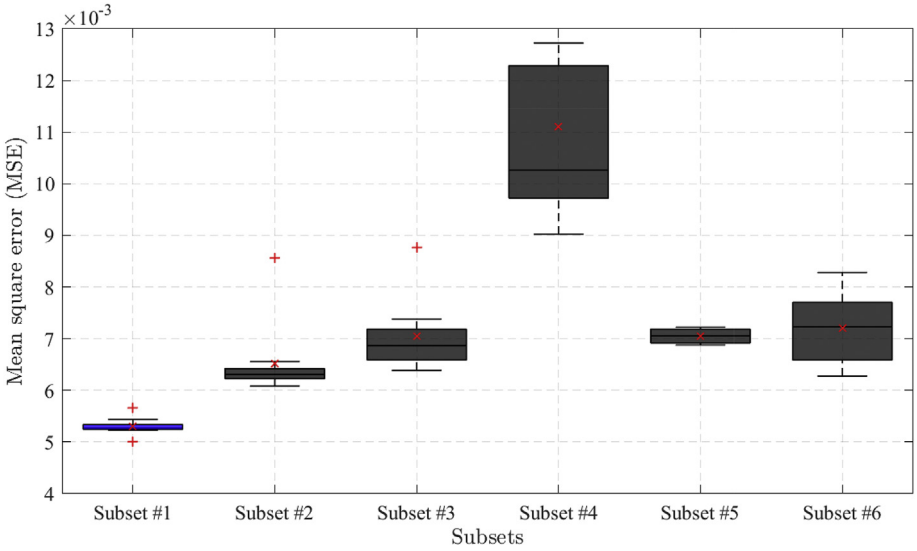

Fig. 8. Subset evaluation based on the standard LSTM model. (a) Plant #1; (b) Plant #2.

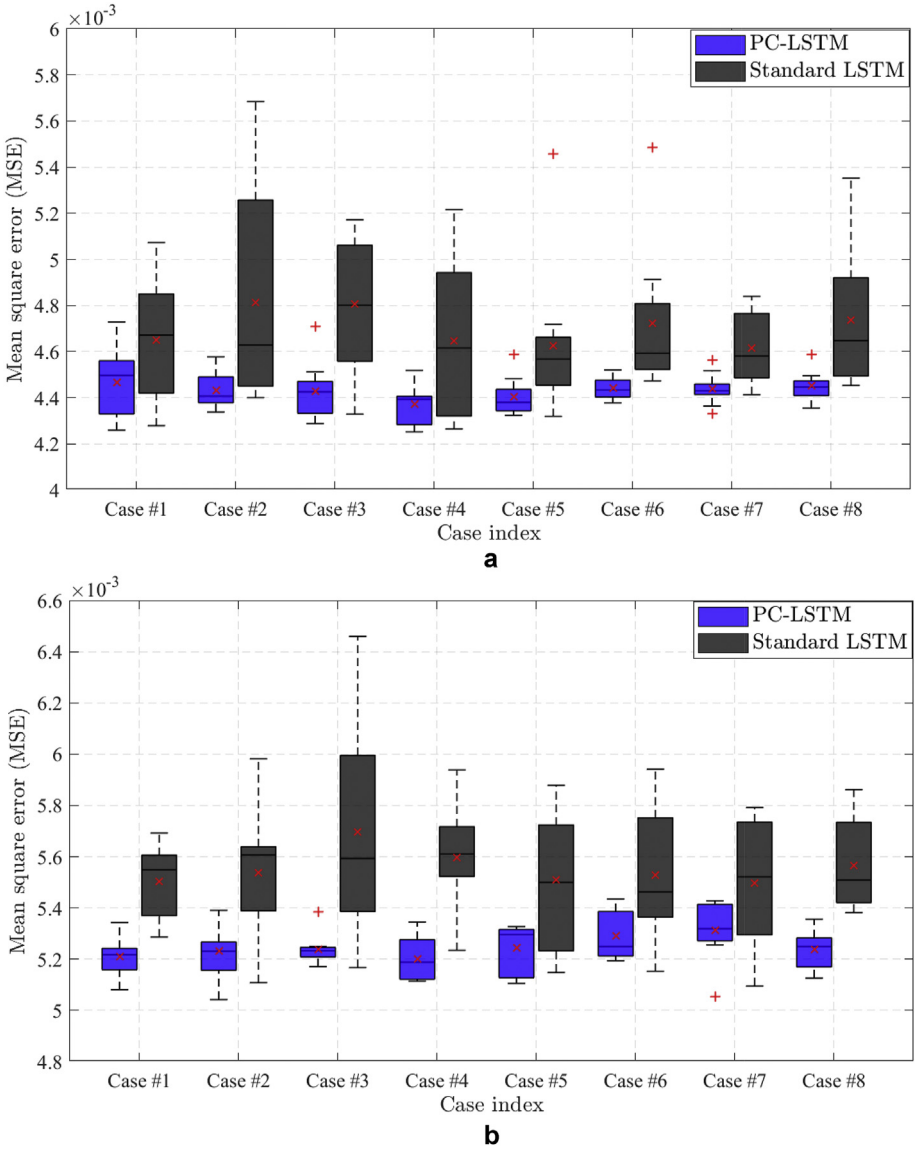

Fig. 9. Robustness comparison between PC-LSTM and LSTM models for two PV plants. (a) Plant #1; (b) Plant #2.

models. The standard deviation (STD), which can be used to describe the stability of the model, is reflected on the bound of boxes. The tighter the bound, the more stable the model. From the results, it is apparent that the proposed PC-LSTM model outperforms the standard LSTM model with higher accuracy (lower averaged MSE) and better stability (tighter bound of box). Specifically, for PV plant #1, the averaged MSEs of PC-LSTM and LSTM model vary from 4.37×10^{-3} (Case #4) to 4.47×10^{-3} (Case #1) and 4.61×10^{-3} (Case #7) to 4.81×10^{-3} (Case #2), respectively. For PV plant #2, similar results can be obtained. The averaged MSEs of PC-LSTM and LSTM vary from 5.20×10^{-3} (Case #4) to 5.31×10^{-3} (Case #7) and 5.50×10^{-3} (Case #7) to 5.70×10^{-3} (Case #3), respectively. Besides, the blue boxes produced by PC-LSTM have tighter bounds than the deep grey boxes produced by LSTM in all cases. Therefore, the obtained results indicate that the proposed PC-LSTM model possesses stronger forecasting capability in terms of robustness than the standard LSTM model.

4.3.2. Forecasting capability comparison between PC-LSTM and standard LSTM - sparse data based

In deep learning, the amount of available data that are used for model training can also affect the model performance. Hence, the forecasting performance of PC-LSTM with sparse data is evaluated. Different amounts (i.e., 20%, 40%, 60%, 80%, and 100%) of the training data from two individual PV plants (same as in Subsection 4.3.1) are adopted to evaluate the sparse data based forecasting capability of the PC-LSTM model.

The comparison results between the proposed PC-LSTM model and the standard LSTM model are illustrated in Fig. 10. From the results, it can be observed that the forecasting error increases with the amount of adopted data decreases for both PV plants, in general, indicating the worth of data in deep learning models. Besides, similar to the results obtained in Subsection 4.3.1, the proposed PC-LSTM model performs much better with higher accuracy (lower averaged MSE) and better stability (tighter bound of box), compared to the standard LSTM model to some extent. The PC-LSTM model can still maintain relatively high accuracy with

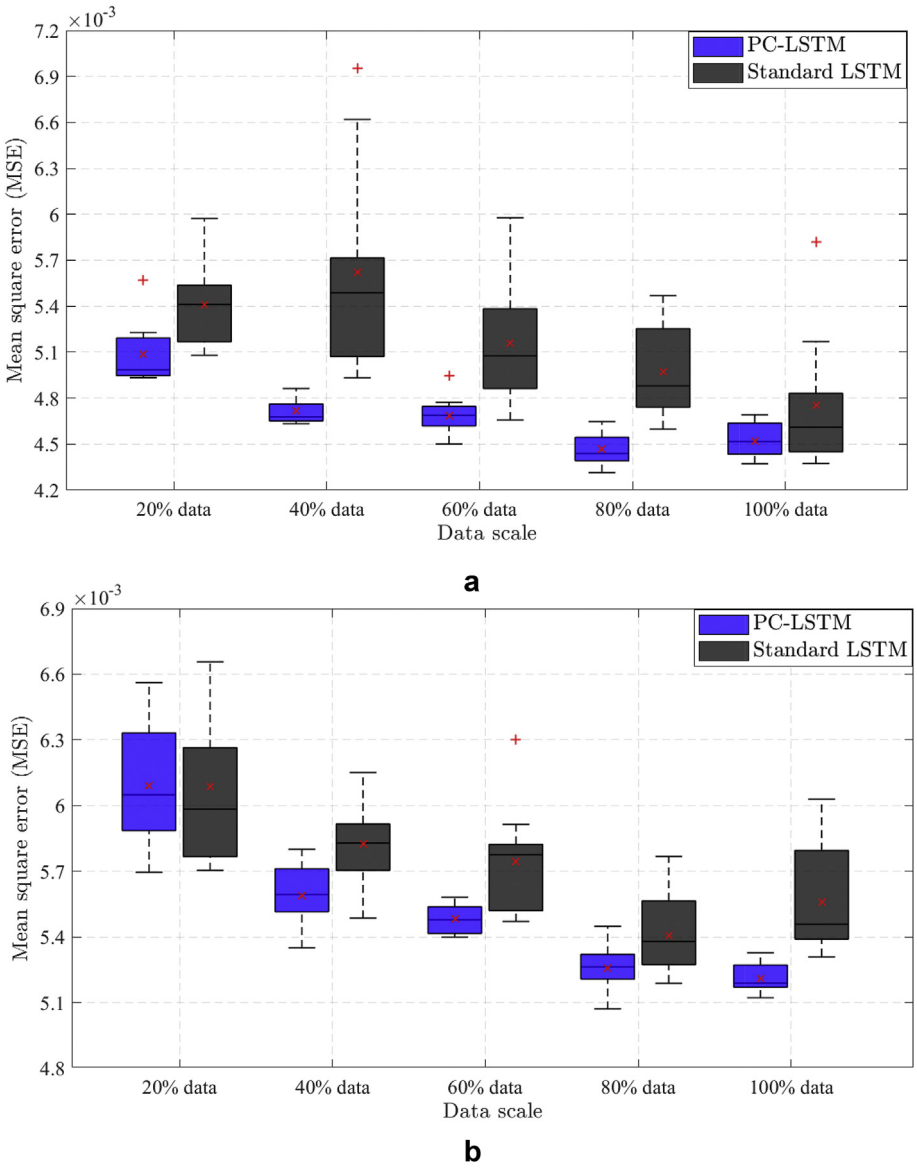

Fig. 10. Comparison between PC-LSTM and LSTM models based on different amounts of data for two PV plants. (a) Plant #1; (b) Plant #2.

sparse data. Therefore, the obtained results illustrate that the PC-LSTM model has stronger capability in terms of sparse data based forecasting and is more suitable for PVPG forecasting with sparse data than the standard LSTM model.

4.4. Forecasting of PV power generation

The performances of hourly day-ahead PVPG forecasting based on the testing datasets of two PV plants by different models are also evaluated. In this case, the proposed PC-LSTM model is mainly compared with the standard LSTM model, the auto-regressive moving average model (ARMA), the k-nearest neighbors regression model (KNN) [18], and the fully-connected neural network (FCNN) with different structures [22–24]. Additionally, a persistence model, which is a forecasting model where the observations of the last day is persisted forward as the forecasting results, is also compared. Other methods, such as XGBoost [39] and Qing's model [40], have already been compared with the standard LSTM model in previous work [5]. The existing results showed the superiority of the standard LSTM model over the literature methods. The same hyper-parameters are adopted for both LSTM and PC-LSTM models. After training the PC-LSTM model multiple times, the suitable hyper-parameters of models, including time step, hidden units, number of layers, learning rate, batch size, are determined for both PV plants as illustrated in Table 4.

To visually display the forecasting results, the hourly day-ahead PVPG forecasting values of seven continuous days of three months

Table 4
Illustration of adopted hyper-parameters within models.

Hyper-parameter	Setting		
	Range	Plant #1	Plant #2
Time step	2–15	4	4
Hidden unit	[32, 64, 96, 128]	64	64
Layer number	[1, 2, 3]	1	1
Learning rate	0.001-0.15	0.05	0.05
Batch size	[64, 128, 256, 512]	512	256

in the testing dataset (April to June, 6 cases in total for two PV plants) are presented as examples in Fig. 11. From the results, it can be seen that all compared models, i.e., ARMA, KNN and FCNN, LSTM, and PC-LSTM, can accomplish the task of PVPG forecasting with acceptable accuracies. The forecasted PVPG curves are in line with the variations of observed PVPG, in general. The result obtained by the PC-LSTM model, however, is smoother and better than the results of the other models to some extent. It is noted that some detailed variations of PVPG are not forecasted well by all methods, since the adopted weather data are forecasted values, which may result in certain difficulty in forecasting. In addition, some physically unreasonable forecasts (e.g., negative power generation) occurred by using the standard LSTM and conventional FCNN without incorporating domain knowledge; whereas, they are significantly restricted by the PC-LSTM model due to the contributions made by the integrated constraints. It is noted that the forecasting of other months can be made with the PC-LSTM model when corresponding weather data are available.

To better understand the results in Fig. 11, the evaluation metrics of above PVPG forecasting examples are presented in Table 5. The best results are marked in bold. The results are also following the previous analysis. The proposed PC-LSTM model can achieve better forecasting performance with lower values of MAE and MSE, and higher values of R^2 score in the comparison in most of cases, which verified the superiority of the PC-LSTM model over other compared models. It is noted that several models can be applied to the PVPG forecasting problem, however, no single model is performing best at all times.

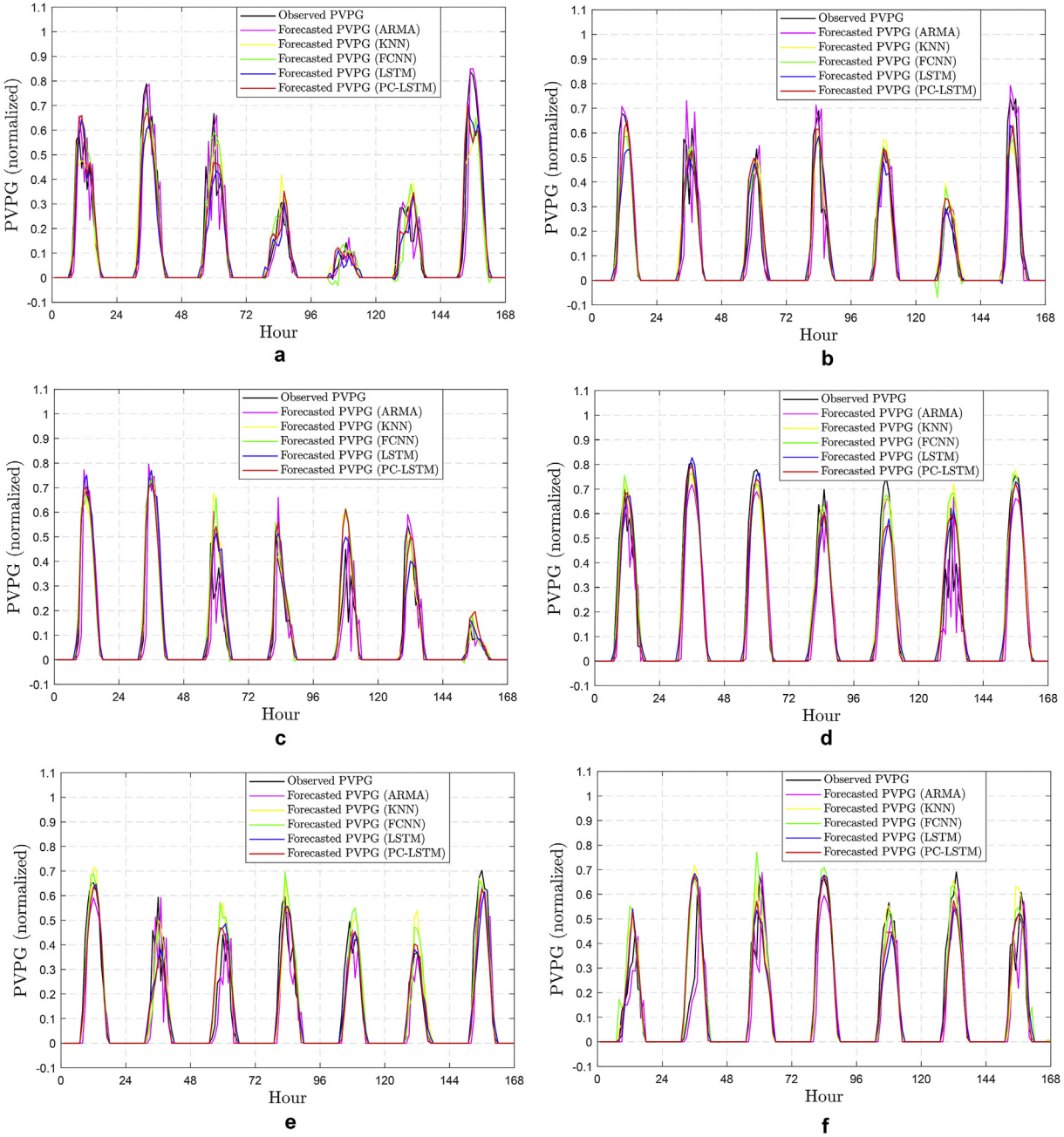

Fig. 11. Examples of visually displayed results of seven days' PVPG forecasting based on different models. (a) Plant #1 - Case in April; (b) Plant #1 - Case in May; (c) Plant #1 - Case in June; (d) Plant #2 - Case in April; (e) Plant #2 - Case in May; (f) Plant #2 - Case in June.

Table 5
Evaluation metrics of displayed PVPG forecasting results by using different models (according to Fig. 11.).

Case index	Models	Plant #1			Plant #2		
		MAE$^{\times 10^{-2}}$	MSE$^{\times 10^{-3}}$	R^2 score	MAE$^{\times 10^{-2}}$	MSE$^{\times 10^{-3}}$	R^2 score
Case in April	ARMA	4.46	8.11	0.815	5.26	9.93	0.862
	KNN	3.64	5.41	0.876	3.13	4.80	0.933
	FCNN	**3.24**	4.72	0.892	**3.03**	4.51	0.937
	LSTM	3.82	5.35	0.878	3.78	5.26	0.926
	PC-LSTM	3.37	**4.60**	**0.895**	3.24	**4.37**	**0.939**
Case in May	ARMA	4.32	9.60	0.777	4.51	8.38	0.790
	KNN	3.56	5.77	0.866	3.81	5.41	0.864
	FCNN	3.15	4.38	0.898	3.59	4.99	0.874
	LSTM	3.39	4.92	0.886	3.40	4.68	0.882
	PC-LSTM	**3.10**	**4.06**	**0.906**	**3.25**	**4.52**	**0.886**
Case in June	ARMA	4.21	10.24	0.697	4.16	7.37	0.831
	KNN	2.69	5.30	0.844	3.79	7.93	0.817
	FCNN	2.67	4.61	0.865	**3.63**	7.30	0.832
	LSTM	2.98	4.29	0.874	3.92	7.38	0.829
	PC-LSTM	**2.62**	**4.20**	**0.876**	3.69	**6.82**	**0.842**

To prevent the impact of randomness, the PVPG forecasting performances for the entire testing dataset by using different models are also evaluated, as illustrated in Table 6. The best results are also marked in bold. From the results, the proposed PC-LSTM model outperforms in all metrics, whereas the ARMA model performs the worst in the overall comparison. Specifically, PC-LSTM and LSTM models outperform KNN and FCNNs with smaller MSE in the evaluation, due to the information from previous time steps within the sequential-data being used in the training of models. Compared to the best MSE result of conventional machine learning methods (FCNN and KNN), about 12.9% and 8.0% improvements of forecasting accuracy can be obtained for PV plant #1 and plant #2, respectively, by using the proposed PC-LSTM model. Additionally, the PC-LSTM model is far more better than the representative of statistical model (ARMA) and the persistence model according to the results. Moreover, compared to the standard LSTM model, the proposed PC-LSTM model performs better to some extent, which indicates that the LSTM model, incorporating some domain knowledge of PV in the form of integrated physical constraints, is more suitable for PVPG forecasting in practice.

4.5. Discussion of results

In the case studies, forecasting capability comparisons between the PC-LSTM model and the standard LSTM model in terms of robustness and sparse data based prediction were proposed at first. The obtained results demonstrated the superiority of PC-LSTM over LSTM due to the integrated physical constraints extracted from domain knowledge of PV. During the training process, the PC-LSTM model was trained with available observations, while being simultaneously guided by some physical constraints. Consequently, the forecasting ability of PC-LSTM can be considerably improved. The proposed PC-LSTM model possesses stronger robustness and is more suitable for PVPG forecasting with sparse data than LSTM.

In addition, the proposed PC-LSTM model was also compared with the conventional machine learning models (i.e., FCNN and KNN), the statistical model (i.e., ARMA), and the persistence model. Specifically, both LSTM and PC-LSTM showed better performance than FCNN and KNN due to the special internal self-looped structure existed in models. Both LSTM and PC-LSTM can preserve previous information and establish temporal correlations between sequential data. However, time-dependence is not considered by FCNN or KNN. Besides, the proposed PC-LSTM model performed much better than the ARMA model and the persistence model, since a variety of weather variables related to PVPG have been considered in the model establishment, whereas the ARMA model and the persistence model were only developed based on the time-series PVPG data. Compared to different models, the obtained results indicate that the PC-LSTM model, when incorporating some domain knowledge of PV, is more suitable for hourly day-ahead PVPG forecasting with higher accuracy in practice.

Although the proposed models performed well in PVPG forecasting, they are data consuming and designed on the basis of the assumption that sufficient training data are available. However, in real-world situations, such as PVPG forecasting for some newly-built PV plants, which only have limited data, this assumption may not hold. Accordingly, the forecasting accuracy of deep learning models cannot be guaranteed. Therefore, how to improve the forecasting performance for PV plants with insufficient data is a significant issue that needs to be solved in future works. In such cases, knowledge transferring or transfer learning between task domains would be a potential solution.

Table 6
Overall evaluation of PVPG forecasting by using different models for the entire testing dataset.

PV plant	Forecasting models	Evaluation metrics		
		MAE$^{\times 10^{-2}}$	MSE$^{\times 10^{-3}}$	R^2 score
Plant #1	Persistence	5.31	16.06	0.660
	ARMA	3.99	8.30	0.824
	KNN	3.33	5.25	0.889
	FCNN (1 hidden layer)	3.13	4.88	0.897
	FCNN (2 hidden layers)	3.27	5.49	0.884
	FCNN (4 hidden layers)	4.40	9.39	0.801
	LSTM	3.23	4.59	0.903
	PC-LSTM	**2.95**	**4.26**	**0.910**
Plant #2	Persistence	5.89	17.85	0.645
	ARMA	4.10	7.02	0.862
	KNN	3.47	5.53	0.890
	FCNN (1 hidden layer)	3.35	5.53	0.890
	FCNN (2 hidden layers)	3.72	7.07	0.860
	FCNN (4 hidden layers)	4.61	10.56	0.790
	LSTM	3.51	5.30	0.895
	PC-LSTM	**3.32**	**5.09**	**0.899**

5. Conclusion

The forecasting of PV power generation has been extremely important throughout the development of the PV industry. This paper proposed an innovative deep learning based framework, and the so-called physics-constrained LSTM, to address the hourly day-

ahead PVPG forecasting problem. The feature variables were processed, and the appropriate feature subset was selected as the model input. Several case studies were illustrated to verify the feasibility and effectiveness of the proposed method. From the results, the proposed method have proved to be successful for producing accurate forecasts of PV power generation, together with realistic prediction intervals. Through PC-LSTM, considering domain knowledge of a specific engineering problem into the construction of ML models is both necessary and meaningful. Deep learning should be not only driven by data, but also by some domain knowledge, which can assist the model to obtain better accuracy, robustness, and interpretability.

In the future, other physical laws, engineering controls, or expert knowledge may be incorporated into the proposed framework, to solve the hourly day-ahead PVPG forecasting problem and other related engineering problems, such as the forecasting of wind power generation. In addition, the proposed framework for incorporating domain knowledge may be combined with other emerging ML algorithms, such as the gated recurrent unit (GRU) [41,42] and the bidirectional encoder representation from transformers (BERT) [43]. Some case studies in the Middle East or other areas of the world also can be carried out in future works.

CRediT author statement

Xing Luo: conceptualization, investigation, methodology, data curation, software, visualization, writing - original draft. **Dongxiao Zhang**: conceptualization, methodology, supervision, writing - review and editing, funding acquisition. **Xu Zhu**: methodology, supervision, writing - review and 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.

Acknowledgements

The research is partially supported with special funds from the Peng Cheng Laboratory and startup funds from the Southern University of Science and Technology (SUSTech).

List of abbreviations

ANN	Artificial neural networks
ARMA	Auto-regressive moving average
DN	Binary feature variable to indicate day and night patterns
ECMWF	European center for medium-range weather forecasts
FCNN	Fully-connected neural network
KNN	K-nearest neighbors
LSTM	Long short-term memory
MAE	Mean absolute error
ML	Machine learning
MSE	Mean square error
PCC	Pearson correlation coefficient
PC-LSTM	Physics-constrained LSTM
PVPG	Photovoltaic power generation
ReLU	Rectified linear unit function
RNN	Recurrent neural network
SGD	Stochastic gradient descent
SR^*	Hourly surface solar radiation downwards
TP^*	Hourly precipitation
TR^*	Hourly thermal radiation downwards
TS^*	Hourly top-net solar radiation

Symbols

Θ	Parameter vector of network
λ_{PLT}	Intensity of penalty on the loss function
P	Value of PCC
Σ	Activation function
$\tilde{C}_t$	Output of the *Tanh* layer at time t
C_t	Updated cell state at time t
$\mathscr{L}(\theta)_{LSTM}$	Loss function of LSTM
$\mathscr{L}(\theta)_{PC-LSTM}$	Loss function of PC-LSTM
MSE_{DATA}	MSE of data loss term
MSE_{PLT}	MSE of penalty loss term
N_{test}	Number of tests
W	Weight parameters of layers
B	Bias parameters of layers
f^{LB}	Lower bound function
f^{UB}	Upper bound function
f_t	Output of the forget gate at time t
i_t	Output of the input gate at time t
o_t	Output of an LSTM block at time t
x_{norm}	Normalized data
x_{orig}	Original data
x_{min}	x_{max} Minimum and maximum values within the dataset.
y_i	i^{th} observed value of PVPG
$\widehat{y_i}$	i^{th} forecasted value of PVPG
$\bar{y}$	Mean of observed values
y_i^*	i^{th} mean value of upper bound and lower bound

Appendix

The analytic forms of the lower and upper bound functions discussed in Section 3.2 can be illustrated in Eqs. (A1) and (A2).

Lower bound function:

$$f^{LB}(x) = \left\{ \frac{p_1^{LB}x^{m-1} + p_2^{LB}x^{m-2} + \cdots + p_m^{LB}}{x^m + q_1^{LB}x^{m-1} + q_2^{LB}x^{m-2} + \cdots + q_m^{LB}} \right\} + \left\{ \xi^{LB} \right\} \quad (A1)$$

Upper bound function:

$$f^{UB}(x) = \left\{ \frac{p_1^{UB}x^{n-1} + p_2^{UB}x^{n-2} + \cdots + p_n^{UB}}{x^n + q_1^{UB}x^{n-1} + q_2^{UB}x^{n-2} + \cdots + q_n^{UB}} \right\} + \left\{ \xi^{UB} \right\} \quad (A2)$$

where x and $f(x)$ represent the value of SR^* and the value of PVPG, respectively; Coefficients p_m^{LB}, p_n^{UB}, q_m^{LB}, and q_n^{UB} of rational functions in the left half of the equations can be easily solved by using the curve fitting toolbox (code: cftool) of MATLAB software; bias ξ^{LB} and ξ^{UB} in the right half of the equations are utilized to slightly adjust the functions; and equation orders m and n are determined according to specific cases.

The bound functions adopted in this work for PV plant #1 and plant #2 can be found in Eqs. (A3)–(A6).

Lower bound function of PV plant #1:

$$f_1^{LB}(x) = \left\{ \frac{157.6x^2 + 288.6x - 10.64}{x^3 - 1418x^2 - 769.1x + 2760} \right\} + \left\{ 3.86e^{-3} \right\} \quad (A3)$$

Upper bound function of PV plant #1:

$$f_1^{UB}(x) = \left\{ \frac{98.15x^2 + 4.471x + 0.03949}{x^3 + 109.3x^2 + 6.552x + 1.447} \right\} + \{0\} \quad (A4)$$

Lower bound function of PV plant #2:

$$f_2^{\mathrm{LB}}(x) = \left\{\frac{434x^3 - 92.98x^2 + 48.99x + 0.7513}{x^3 + 118.8x^2 - 138.3x + 554.7}\right\} - \left\{1.35e^{-3}\right\} \tag{A5}$$

Upper bound function of PV plant #2:

$$f_2^{\mathrm{UB}}(x) = \left\{\frac{4574x^2 + 6031x - 9.602}{x^3 + 3332x^2 - 7036x + 408.5}\right\} + \left\{2.35e^{-2}\right\} \tag{A6}$$

References

[1] Das UK, Tey KS, Seyedmahmoudian M, Mekhilef S, Idris MYI, Deventer WV, Horan B, Stojcevski A. Forecasting of photovoltaic power generation and model optimization: a review. Renew Sustain Energy Rev 2018;81:912–28. https://doi.org/10.1016/j.rser.2017.08.017.

[2] Zhao X, Zhang Y. Technological progress and industrial performance: a case study of solar photovoltaic industry. Renew Sustain Energy Rev 2018;81:929–36. https://doi.org/10.1016/j.rser.2017.08.038.

[3] Sayigh A. Renewable energy - the way forward. Appl Energy 1999;64(1):15–30. https://doi.org/10.1016/S0306-2619(99)00117-8.

[4] Gueymard C. The sun's total and spectral irradiance for solar energy applications and solar radiation models. Sol Energy 2004;76(4):423–53. https://doi.org/10.1016/j.solener.2003.08.039.

[5] Zheng J, Zhang H, Dai Y, Wang B, Zheng T, Liao Q, Liang Y, Zhang F, Song X. Time series prediction for output of multi-region solar power plants. Appl Energy 2020;257:114001. https://doi.org/10.1016/j.apenergy.2019.114001.

[6] Lin S, Li C, Xu F, Liu D, Liu J. Risk identification and analysis for new energy power system in China based on d numbers and decision-making trial and evaluation laboratory (dematel). J Clean Prod 2018;180:81–96. https://doi.org/10.1016/j.jclepro.2018.01.153.

[7] Strzalka A, Alam N, Duminil E, Coors V, Eicker U. Large scale integration of photovoltaics in cities. Appl Energy 2012;93:413–21. https://doi.org/10.1016/j.apenergy.2011.12.033.

[8] Dolara A, Leva S, Manzolini G. Comparison of different physical models for pv power output prediction. Sol Energy 2015;119:83–99. https://doi.org/10.1016/j.solener.2015.06.017.

[9] Koster D, Minette F, Braun C, O'Nagy O. Short-term and regionalized photovoltaic power forecasting enhanced by reference systems on the example of Luxembourg. Renew Energy 2019;132:455–70. https://doi.org/10.1016/j.renene.2018.08.005.

[10] Burnham K, Anderson D, Huyvaert KK. Aic model selection and multimodel inference in behavioral ecology: some background, observations, and comparisons. Behav Ecol Sociobiol 2011;65:23–35. https://doi.org/10.1007/s00265-010-1084-z.

[11] Boland J, David M, Lauret P. Short term solar radiation forecasting: island versus continental sites. Energy 2016;113:186–92. https://doi.org/10.1016/j.energy.2016.06.139.

[12] Reikard G. Predicting solar radiation at high resolutions: a comparison of time series forecasts. Sol Energy 2009;83(3):342–9. https://doi.org/10.1016/j.solener.2008.08.007.

[13] Li Y, Su Y, Shu L. An armax model for forecasting the power output of a grid connected photovoltaic system. Renew Energy 2014;66:78–89. https://doi.org/10.1016/j.renene.2013.11.067.

[14] Zhang X, Fang F, Liu J. Weather-classification-mars-based photovoltaic power forecasting for energy imbalance market. IEEE Trans Ind Electron 2019;66(11):8692–702. https://doi.org/10.1109/TIE.2018.2889611.

[15] Wang G, Su Y, Shu L. One-day-ahead daily power forecasting of photovoltaic systems based on partial functional linear regression models. Renew Energy 2016;96:469–78. https://doi.org/10.1016/j.renene.2016.04.089.

[16] Sheng H, Xiao J, Cheng Y, Ni Q, Wang S. Short-term solar power forecasting based on weighted Gaussian process regression. IEEE Trans Ind Electron 2018;65(1):300–8. https://doi.org/10.1109/TIE.2017.2714127.

[17] Trapero JR, Kourentzes N, Martin A. Short-term solar irradiation forecasting based on dynamic harmonic regression. Energy 2015;84:289–95. https://doi.org/10.1016/j.energy.2015.02.100.

[18] Zhang Y, Wang J. K-nearest neighbors and a kernel density estimator for gefcom2014 probabilistic wind power forecasting. Int J Forecast 2016;32(3):1074–80. https://doi.org/10.1016/j.ijforecast.2015.11.006.

[19] Brown RG. Exponential smoothing for predicting demand. Little; 1956.

[20] Dev S, AlSkaif T, Hossari M, Godina R, Louwen A, van Sark W. Solar irradiance forecasting using triple exponential smoothing. In: International Conference on smart energy systems and technologies; 2018. p. 1–6. https://doi.org/10.1109/SEST.2018.8495816.

[21] Mueller JP, Massaron L. Machine learning for dummies. For Dummies; 2016.

[22] Abuella M, Chowdhury B. Solar power forecasting using artificial neural networks. In: North American Power Symposium; 2015. p. 1–5. https://doi.org/10.1109/NAPS.2015.7335176.

[23] Liu J, Fang W, Zhang X, Yang C. An improved photovoltaic power forecasting model with the assistance of aerosol index data. IEEE Transactions on Sustainable Energy 2015;6(2):434–42. https://doi.org/10.1109/TSTE.2014.2381224.

[24] Manjili YS, Vega R, Jamshidi MM. Data-analytic-based adaptive solar energy forecasting framework. IEEE Systems Journal 2018;12(1):285–96. https://doi.org/10.1109/JSYST.2017.2769483.

[25] Müller KR, Smola AJ, Rätsch G, Schölkopf B, Kohlmorgen J, Vapnik V. Predicting time series with support vector machines. In: Gerstner Wulfram, Germond Alain, Hasler Martin, Nicoud J Daniel, editors. Artificial neural networks. Berlin, Heidelberg: Springer Berlin Heidelberg; 1997. p. 999–1004. https://doi.org/10.1007/BFb0020283.

[26] Tang P, Chen D, Hou Y. Entropy method combined with extreme learning machine method for the short-term photovoltaic power generation forecasting. Chaos, Solit Fractals 2016;89:243–8. https://doi.org/10.1016/j.chaos.2015.11.008. nonlinear Dynamics and Complexity.

[27] Cervone G, Harding LC, Alessandrini S, Monache LD. Short-term photovoltaic power forecasting using artificial neural networks and an analog ensemble. Renew Energy 2017;108:274–86. https://doi.org/10.1016/j.renene.2017.02.052.

[28] Eseye AT, Zhang J, Zheng D. Short-term photovoltaic solar power forecasting using a hybrid wavelet-pso-svm model based on scada and meteorological information. Renew Energy 2018;118:357–67. https://doi.org/10.1016/j.renene.2017.11.011.

[29] Wang F, Zhang Z, Liu C, Yu Y, Pang S, Duic N, Khah MS, Catalao JP. Generative adversarial networks and convolutional neural networks based weather classification model for day ahead short-term photovoltaic power forecasting. Energy Convers Manag 2019;181:443–62. https://doi.org/10.1016/j.enconman.2018.11.074.

[30] Schuster M, Paliwal KK. Bidirectional recurrent neural networks. IEEE Trans Signal Process 1997;45(11):2673–81. https://doi.org/10.1109/78.650093.

[31] Gensler A, Henze J, Sick B, Raabe N. Deep learning for solar power forecasting - an approach using autoencoder and lstm neural networks. In: IEEE International Conference on Systems 2. Man, and Cybernetics; 2016. p. 2858–65. https://doi.org/10.1109/SMC.2016.7844673.

[32] Mohamed AN, Karar M. Accurate photovoltaic power forecasting models using deep lstm-rnn. Neural Comput Appl 2017:2727–40. https://doi.org/10.1007/s00521-017-3225-z.

[33] Wang N, Zhang D, Chang H, Li H. Deep learning of subsurface flow via theory-guided neural network. J Hydrol 2020;584:124700. https://doi.org/10.1016/j.jhydrol.2020.124700.

[34] Dash M, Liu H. Feature selection for classification. Intell Data Anal 1997;1(1):131–56. https://doi.org/10.1016/S1088-467X(97)00008-5.

[35] Xu H, Deng Y. Dependent evidence combination based on shearman coefficient and pearson coefficient. IEEE Access 2018;6:11634–40. https://doi.org/10.1109/ACCESS.2017.2783320.

[36] von Rueden L, Mayer S, Beckh K, Georgiev B, Giesselbach S, Heese R, Kirsch B, Pfrommer J, Pick A, Ramamurthy R, Walczak M, Garcke J, Bauckhage C, Schuecker J. Informed machine learning – a taxonomy and survey of integrating knowledge into learning systems. 2019. arXiv:1903.12394.

[37] Luo X, Zhu X, Lim EG. A parametric bootstrap algorithm for cluster number determination of load pattern categorization. Energy 2019;180:50–60. https://doi.org/10.1016/j.energy.2019.04.089.

[38] Hong T, Pinson P, Fan S, Zareipour H, Troccoli A, Hyndman RJ. Probabilistic energy forecasting: global energy forecasting competition 2014 and beyond. Int J Forecast 2016;32(3):896–913. https://doi.org/10.1016/j.ijforecast.2016.02.001.

[39] X. Yu, Comparison of support vector machine and extreme gradient boosting for predicting daily global solar radiation using temperature and precipitation in humid subtropical climates: a case study in China, Energy Convers Manag 164. doi:10.1016/j.enconman.2018.02.087.

[40] Qing X, Niu Y. Hourly day-ahead solar irradiance prediction using weather forecasts by lstm. Energy 2018;148:461–8. https://doi.org/10.1016/j.energy.2018.01.177.

[41] Li W, Logenthiran T, Woo WL. Multi-gru prediction system for electricity generation's planning and operation, IET Generation. Transm Distrib 2019;13(9):1630–7. https://doi.org/10.1049/iet-gtd.2018.6081.

[42] Deng Y, Wang L, Jia H, Tong X, Li F. A sequence-to-sequence deep learning architecture based on bidirectional gru for type recognition and time location of combined power quality disturbance. IEEE Transactions on Industrial Informatics 2019;15(8):4481–93. https://doi.org/10.1109/TII.2019.2895054.

[43] Devlin J, Chang M, Lee K, Toutanova K. Bert: pre-training of deep bidirectional transformers for language understanding. 2018. arXiv:1810.04805.