

Multi-view Deep Forecasting for Hourly Solar Irradiance with Error Correction

Cankun Zhong ^a, Chun Sing Lai ^{b,c*}, Wing W. Y. Ng ^{a,*}, Yingshan Tao ^c, Ting Wang ^a, Loi Lei Lai ^{c,*}

^a Guangdong Provincial Key Laboratory of Computational Intelligence and Cyberspace Information, School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China

^b Brunel Interdisciplinary Power Systems Research Centre, Department of Electronic and Electrical Engineering, Brunel University London, London, UB8 3PH, UK

^c Department of Electrical Engineering, School of Automation, Guangdong University of Technology, Guangzhou 510006, China

* Corresponding authors: chunsing.lai@brunel.ac.uk (C. S. Lai); [W. W. Y. Ng \(wingng@ieee.org\);](mailto:W. W. Y. Ng (wingng@ieee.org);)
l.l.lai@ieee.org (L. L. Lai)

Abstract

Short-term solar irradiance forecasting is crucial in managing power network operations and solar photovoltaic applications. In this paper, a Multi-view Deep Forecasting method with Error Correction (MvDF_EC) for 1-hour ahead solar forecasting is proposed. MvDF_EC comprises of the Multi-view Deep Forecasting method (MvDF) and a robust Radial Basis Function Neural Network trained via minimizing the Localized Generalization Error for compensating the solar forecasting error of MvDF. MvDF consists of three deep neural networks which learn representations of input data from different views. The three views are 1) the hierarchical local temporal information extracted by the Temporal Convolutional Neural Network (TCN), 2) the key context sequential information captured by the Bi-directional Long Short-Term Memory Neural Network with Temporal Attention (BLSTMattn), and 3) long-term temporal dependencies between local temporal patterns filtered by the Convolutional Gated Recurrent Unit Neural Network (C_GRU). The solar forecasting performance of the proposed MvDF_EC is evaluated with the National Solar Radiation Database. Simulation results show that MvDF_EC yields the most accurate solar

prediction compared with the benchmarks including the smart persistence and the state-of-the-art models. The lowest relative Root Mean Square Error values for Maraba and Labelle are 22.08% and 27.40%, respectively in 1-hour ahead solar forecasting.

Keywords: Solar forecasting, multi-view deep forecasting, error correction

1. Introduction

Forecasting of renewable energy resources such as wind and solar is important for the optimal operation of low-carbon power systems (Su. et al., 2020; Wu. et al., 2020). Solar energy is regarded as one of the cleanest forms of energy and the short-term solar forecasting is crucial in managing power network operations (Huang et al., 2018; Lai. et al., 2017a). The real-life Global Horizontal Irradiance (GHI) predictions can address the performance characteristics of solar photovoltaic applications. But due to the influence of cloud amount and atmospheric state, the short-term forecasting is challenging.

In previous studies, many data-driven approaches were proposed for short-term solar forecasting. These approaches can be mainly divided into three parts: the physical methods, the statistical methods, and the machine learning methods. The physical methods are based on mathematical meteorological data prediction (Larson et al., 2016) and the statistical methods utilizing the real historical solar time series data for forecasting include the autoregressive integrated moving average and the Markov Chain model (Jiang et al., 2017), among others. Recently, machine learning based approaches have been paid more attention, including the unsupervised clustering for de-trending solar data, adaptive learning, and deep learning.

A novel clustering method TB_K-means (Azimi et al., 2016) was proposed to partition the solar data into several clusters where the solar data in each cluster is more stationary and more predictable. The TB_K-means is combined with the multiple layer perceptron to develop the hybrid hourly solar radiation forecasting method. The similar hybrid forecasting method was proposed (Feng et al., 2018), where multiple machine learning models were adopted including the Support Vector Machine (Bae et al., 2017), the Artificial Neural Network (Rodríguez et al., 2018), and the Random Forest (Benali et al., 2019) to forecast the hourly real-life GHI of each cluster. The clustering technique is utilized in a different way (Sun et al., 2018) where the K-means clustering is used to cluster the forecasting results of each sub-component generated from the Ensemble Empirical Mode Decomposition method, and a least squares support vector regression is applied to ensemble the sub-component forecasts of each cluster.

A novel adaptive learning hybrid model was proposed (Wang et al., 2018) which consists of a time-varying multiple-linear model for capturing the linear relationship, a neural

network for capturing the non-linear one, and an adaptive learning online algorithm for improving the predicting performance adaptively online. The study (Sanfilippo et al., 2016) proposed a novel multi-modeling approach for solar forecasting. The supervised classification of forecasting evaluation results from diverse models was used to select the best predictions, according to their expected superiority in terms of lower error rate.

Deep learning had been applied widely to the renewable energy, such as wind power (Li, et al., 2018; Li et al., 2019) and solar energy (Feng and Zhang, 2020; Zhen et al., 2020). The Convolutional Neural Network (CNN) was adopted (Feng and Zhang, 2020) to forecast the solar PV output using the contemporaneous images of the sky, while the study (Zhen et al., 2020) firstly assigned the sky image to the corresponding class using the deep clustering method and then utilized a corresponding hybrid deep learning method for PV power forecasting. The meteorological features were utilized (Qing and Niu, 2018) as the input for a Long Short-Term Memory neural network (LSTM) for day-ahead hourly solar radiance prediction. Furthermore, the study (Ghimire et al., 2019) applied CNN to extract features from predictive variables while the LSTM absorbed them for solar radiation prediction. In contrast, the study (Zhang et al., 2020) adopted the CNN in a different way where the CNN was used to extract spatial correlation from meteorological parameters associated with a target site and its neighboring sites. Besides, a LSTM is applied to extract the temporal features from historical solar irradiance data and the correlations of the CNN and LSTM are merged finally to predict the solar irradiance.

Different types of deep learning models such as CNN and LSTM can capture a different type of temporal information. The CNN is conducted to get the local temporal information of the input sequences (i.e., GHI feature sequences and meteorological feature sequences) while the LSTM can learn the long-term dependency of the input sequences. Thus, in this study a Multi-view Deep Forecasting with Error Correction (MvDF_EC) method is proposed for 1-hour ahead real-life GHI forecasting. MvDF_EC utilizes the idea of multi-view learning (Zhang et al., 2016) to describe the input sequences more comprehensively and accurately in different ways using three different deep neural networks. They are the Temporal Convolutional Neural Network (TCN) (Bai et al., 2018), the Bi-directional Long Short-term Memory Neural Network with Temporal Attention (BLSTMAttN) (Bin et al., 2019), and the Convolutional Gated Recurrent Unit Neural Network (C_GRU) (Zuo et al., 2016). Furthermore, an error correction method is employed in MvDF_EC to obtain higher forecasting accuracy. The major contributions of this work are as follows:

- 1) A Multi-view Deep Forecasting (MvDF) method is proposed for 1-hour ahead real-life GHI prediction by exploiting three different views learned by three deep neural networks. The three views are 1) the hierarchical local temporal information extracted by the TCN, 2) the key context sequential information captured by BLSTMAttN, and 3) long-term temporal dependencies between local temporal patterns filtered by the C_GRU. These three views are then combined by the complementary and the consensus principles for training MvDF. To the best knowledge of the authors, this is the first multi-view deep learning work for the solar radiance forecasting problem.
- 2) The Radial Basis Function Neural Network (RBFNN) trained via minimizing the Localized Generalization Error (L-GEM) (Yeung et al., 2007) is adopted to compensate

for the prediction error of MvDF. Instead of performing the forecasting, the RBFNN predicts the error made by MvDF using the same input. L-GEM minimizes the generalization error made for unseen samples similar to the training samples only such that minor errors made by MvDF are compensated. However, L-GEM ignores very different unseen samples to allow a large error for very different unseen samples which are learned in training and expected to be correct.

- 3) Extensive simulations are carried out to confirm the superiority of the proposed method. Simulation results on the National Solar Radiation Database show that the proposed method outperforms existing methods for solar forecasting in most cases.

The remainder of this paper is organized as follows. Section 2 describes the proposed method for 1-hour ahead real-life GHI forecasting. Section 3 gives the simulation results and detailed discussion. Finally, we provide the conclusion and future works in Section 4.

2. Multi-view Deep Forecasting with Error Correction for Real-life GHI Forecasting

In this study, the proposed MvDF_EC consists of MvDF and RBFNN. MvDF is proposed for 1-hour ahead real-life GHI forecasting, while RBFNN trained via minimization of L-GEM is used to predict the forecasting error of MvDF. The final forecast result of MvDF is the summation of outputs of MvDF and RBFNN. Both MvDF and RBFNN take the same input features and the features are from two information sources, namely (i) GHI features: real-life GHI, clear-sky GHI, clear-sky index, and solar zenith angle. (ii) Meteorological features: temperature, relative humidity, wind speed, wind direction, and pressure. Note that the clear-sky GHI is calculated from the solar model without considering the environmental effect but the real-life GHI is moreover affected by the weather conditions (Lai. et al., 2017b). The overall forecasting procedure of MvDF_EC is shown in Fig. 1.

In the following section, we first introduce three representative deep neural networks for time series prediction in Section 2.1, which are used to generate different representations from the same input sequences. Then, the proposed MvDF is given in Section 2.2. Finally, the robust RBFNN for error correction of MvDF is given in Section 2.3.

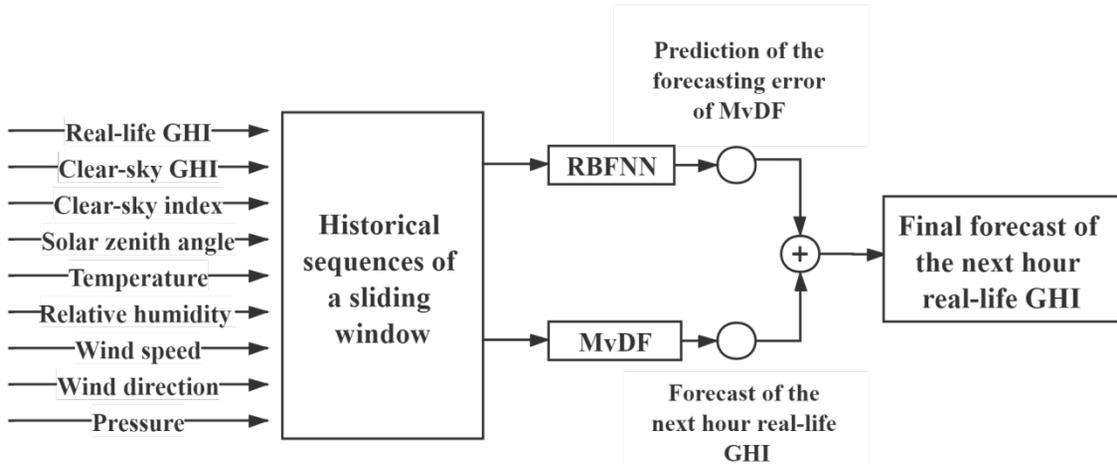


Fig. 1. 1-hour ahead real-life GHI forecasting procedure of MvDF_EC.

2.1. Multi-view Construction

Three representative types of deep learning models for time series prediction (Han et al., 2019) are adopted in this paper to provide three different views. More specifically, they are TCN, BLSTMattn, and C_GRU. All these three deep neural networks capture the temporal features in different ways. Each layer of TCN learns the short-term relationship of the input sequences. The long-term useful temporal representations for real-life GHI forecasting are established by layer-by-layer stacking the convolutional layers. BLSTMattn learns the context information through BLSTM and focuses on key temporal information for real-life GHI forecasting by the temporal attention mechanism. C_GRU utilizes CNN as the filter for extracting higher-level local temporal patterns which are fed into GRU to learn the representations for real-life GHI forecasting.

In fact, these three models have their own limitations. As for TCN, in the case of its local kernel and layer-by-layer stacking characteristics, the information of those closer to the predicted time may not well focused while this information is important for real-life GHI forecasting. On the contrary, BLSTMattn can more easily remember the recent information of the input sequences, but it cannot solve the vanishing gradients problem completely. Although the hybrid model C_GRU can solve the drawbacks of the two above mentioned models to some extent, it easily gets affected by different random weight initializations due to the more complex model architecture and hence behaves unstable. Therefore, the multi-view learning framework is adopted in this study to utilize the knowledge from these three models (views) to establish a more effective model for real-life GHI forecasting. Brief introductions of these three deep neural networks are given as follows:

1) Temporal Convolutional Neural Network (TCN)

TCN adopts the 1D dilated causal convolution in each layer to capture the local temporal patterns. The 1D dilated causal convolution is implemented with the left zero padding such that the output of each layer at time t is convolved only with features from time t and earlier in the previous layer. Besides, it supports an exponential expansion of the receptive field without loss of coverage, which enables a longer time dependency to be captured. Suppose x^{l-1} is an input of the l^{th} convolutional network layer consisting of multiple dilated causal convolutions, and the convolutional kernel of l^{th} layer is $C=\{f_1, f_2 \dots f_w\}$, then the output of the l^{th} convolutional layer at time step t is defined as:

$$C(t) = \sum_{i=1}^k f_i x_{t-d(k-i)}^{l-1} \quad (1)$$

where d is the dilation factor and k is the kernel size. By stacking the convolutional layers, more past information can be fused in the deeper layer and thus more high-level temporal features can be learned. In addition, the identity mapping (Li and Wang, 2017) is also adopted in TCN to stabilize a deeper and larger network. The illustration of a residual block is given in Fig. 2 below:

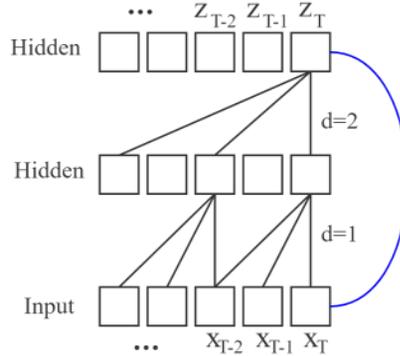


Fig. 2. A residual block of TCN. The black lines are kernels and the blue line indicates the identity mapping.

2) Bi-directional Long Short-Term Memory Neural Network with Temporal Attention (BLSTMattn)

BLSTMattn adopts the Bi-directional Long Short-term Neural Network (BLSTM) to capture the temporal relation of the input sequences. Besides, the temporal attention mechanism is used in BLSTMattn to explicitly enhance the importance of different hidden states at different time. The overall architecture of BLSTMattn is shown in Fig. 3.

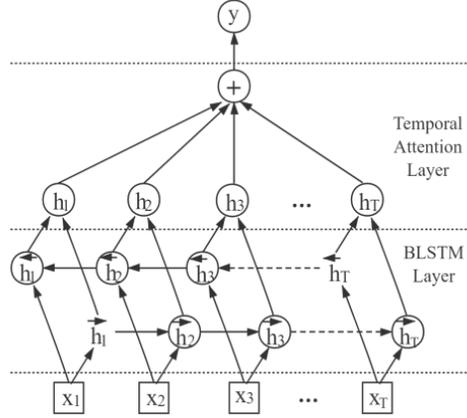


Fig. 3. The architecture of BLSTMattn.

BLSTM consists of two unidirectional LSTMs with opposite directions so that it can capture the context information. The LSTM consists of four basic components: the input gate i_t , the forget gate f_t , the output gate o_t , and the cell state c_t . The forget gate f_t takes current input x_t and previous hidden state h_{t-1} as inputs to determine how much information of the previous cell state c_{t-1} will be forgotten using the sigmoid function $\sigma(\cdot)$. x_t and h_{t-1} are also used for calculating a new candidate cell state \tilde{c}_t . The input gate i_t determines which information of \tilde{c}_t should be updated into the stored cell state c_t . The output gate o_t controls how much information c_t is given and thus results in the hidden state h_t as demonstrated by the following equations:

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

$$\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (3)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (4)$$

$$c_t = f_t * c_{t-1} + i_t * \tilde{c}_t \quad (5)$$

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

$$h_t = o_t * \tanh(c_t) \quad (7)$$

where \cdot denotes the dot product and $*$ denotes the element-wise product. W_j and b_j ($\forall j \in \{i, f, o, \tilde{c}\}$) are learnable parameters.

BLSTM concatenates two hidden states of opposite directions at the same time t as the final hidden state:

$$h_t = [\vec{h}_t, \bar{h}_t] \quad (8)$$

where \vec{h}_t and \bar{h}_t denote the t^{th} hidden states of two LSTMs with forward and past-ward direction, respectively.

The temporal attention mechanism is applied to the hidden states to denote the importance of each h_t by the corresponding attention weight α_t . The weighted sum of all hidden states is denoted as r_H which is used for real-life GHI forecasting:

$$u_t = \tanh(h_t) \quad (9)$$

$$\alpha_t = \frac{\exp(u_t \cdot u_w)}{\sum_t \exp(u_t \cdot u_w)} \quad (10)$$

$$r_H = \sum_t \alpha_t * h_t \quad (11)$$

where u_w is a learnable parameter vector.

3) Convolutional Gated Recurrent Unit Neural Network (C_GRU)

As shown in Fig. 4, the C_GRU adopts convolutional layers to learn the local temporal features of the input sequences while the Gated Recurrent Unit neural network (GRU) (Zhao et al., 2018) takes the local temporal patterns to learn the long-term dependencies.

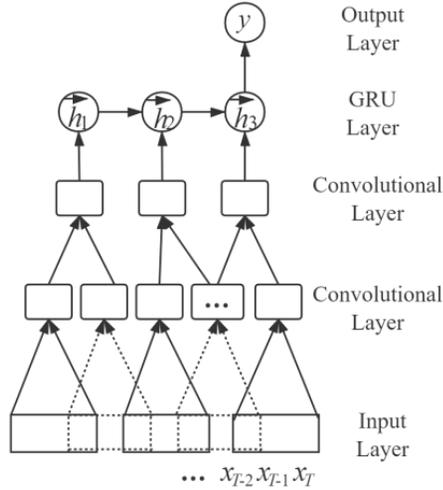


Fig. 4. The illustration of the C_GRU.

GRU consists of the reset gate, update gate, and the hidden state. The reset gate r_t is used to decide whether to ignore the previous hidden state h_{t-1} . The update gate z_t is used to determine how much memories should be updated by a new candidate hidden state \tilde{h}_t , as shown in the following equations:

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r) \quad (12)$$

$$\tilde{h}_t = \tanh(W_{\tilde{h}} \cdot [r_t * h_{t-1}, x_t] + b_{\tilde{h}}) \quad (13)$$

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z) \quad (14)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \quad (15)$$

2.2. Multi-view Deep Forecasting (MvDF)

Considering a single deep neural network has its own characteristic and limitation to capture the temporal information of the input sequences, three deep neural networks (TCN, BLSTMattn, and C_GRU) are adopted to consider the temporal characteristics of the same input sequences in different perspectives to describe the data more comprehensively and accurately. The multi-view learning paradigm is utilized to make full use of the information provided by each deep neural network. The success of multi-view learning strongly depends on complementary and consensus principles.

1) Complementary Principle

The complementary principle states that utilizing different representations of the data can describe the data more accurately since each representation may contain some information that others do not know. In this paper, three different deep neural networks forecast the next hour real-life GHI by using extracted representations of their own. The view attention is utilized to combine the forecasting outputs of three deep neural networks appropriately under the complementary principle. Let O denotes the matrix consisting of outputs of three networks. That is $O=[o_1, o_2, \dots, o_V]$ and the number of views $V=3$. Let $W_v = [w_1, w_2, \dots, w_V]$ where w_v ($v=1, 2, \dots, V$) denotes the weight of the output of the v^{th} deep neural network. The following equations show how to calculate the view weight matrix W_v :

$$M = \tanh(W_w \cdot O) \quad (16)$$

$$W_v = \text{softmax}(u_v \cdot M) \quad (17)$$

where W_w and u_v are learnable parameters. Therefore, the complementary loss is established as shown in the following:

$$\text{complementary loss} = \sum_{n=1}^N \left\| \left(\sum_{v=1}^V w_v f_v(s_n) \right) - y_n \right\|_2^2 \quad (18)$$

where $f_v(\cdot)$, s_n , y_n , and N represent the v^{th} deep neural network, the n^{th} sample, the label of the corresponding n^{th} sample, and the total number of samples, respectively.

2) Consensus Principle

The consensus principle aims at minimizing the disagreement on multiple distinct views. The study reported (Dasgupta et al., 2002) proves that the probability of the disagreement of two independent hypotheses f_1 and f_2 on unseen samples is the upper bound of the error rate of each hypothesis, as shown in the following:

$$P(f_1 \neq f_2) \geq \max(P_{\text{error}}(f_1), P_{\text{error}}(f_2)) \quad (19)$$

Thus, by minimizing the disagreement probability of two hypotheses on unseen samples, the forecasting accuracy of each hypothesis can be improved. In this paper, the consensus loss function is also established to minimize the disagreement of three deep neural networks' forecasting outputs on the perturbed input samples \tilde{s}_n which are generated by adding small Gaussian noises to the original samples s_n :

$$\text{consensus loss} = \sum_{n=1}^N \sum_{i=1, i < j}^V \|f_i(\tilde{s}_n) - f_j(\tilde{s}_n)\|_2^2 \quad (20)$$

Therefore, the final objective function for training the proposed MvDF model is given by minimizing both the complementary loss and consensus loss:

$$\min \sum_{n=1}^N \left\| \left(\sum_{v=1}^V w_v f_v(s_n) \right) - y_n \right\|_2^2 + \lambda \sum_{n=1}^N \sum_{i=1, i < j}^V \|f_i(\tilde{s}_n) - f_j(\tilde{s}_n)\|_2^2 \quad (21)$$

where λ is the hyper-parameter to balance the complementary and consensus principles.

It should be noted that MvDF is not trained from scratch in this study. This is because a deep neural network with randomly initialized weights may fail to accomplish the difficult real-life GHI forecasting task and thus each network cannot provide useful information to each other to improve the forecasting accuracy. Therefore, each deep neural network is pre-trained with the labeled data first. Then, as shown in Fig. 5, they are jointly fine-tuned according to Equation (21). The blue color indicates the data flow for calculating the complementary loss while the red color indicates the data flow for calculating the consensus loss. The circles represent deep networks' outputs. The output of the View Attention is the view weight matrix W_v .

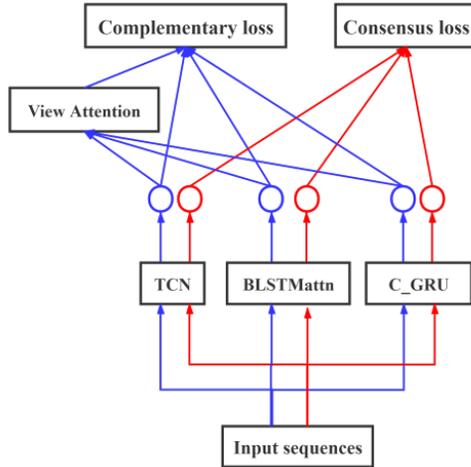


Fig. 5. Fine-tuning stage of MvDF.

2.3. Error Correction for MvDF (MVDF_EC)

Since the proposed MvDF may still suffer from systematic errors (Liu et al., 2019) (e.g., making a too high or too low prediction), RBFNN trained via minimizing the L-GEM is utilized to further enhance the real-life GHI forecasting accuracy. More specifically, RBFNN takes the same input as MvDF but tries to predict the next hour real-life GHI forecasting error generated from MvDF rather than the next hour real-life GHI values. In this work, the historical forecasting error is not severed as the input feature to the RBFNN because the possible different distribution between the training and testing input features to MvDF may enlarge the forecasting error distribution difference between the training and testing phase. If MvDF fails to generalize well to the unseen testing data which follows a slightly different distribution compared with the training data, the RBFNN utilizing the training forecasting error of MvDF as input feature may not perform well when taking the testing forecasting error of MvDF as input.

The purpose of RBFNN training is to find a network structure and connection weight to minimize the generalization error. In fact, once the number of hidden neurons is determined, centers and widths of hidden neurons can be obtained by K-means clustering. After fixing both centers and widths, connection weights can be calculated by a pseudo-inverse technique. Therefore, the objective of RBFNN training can be simplified to the finding of the optimal number of hidden neurons which minimizes the generalization error. We cannot, however, directly estimate the generalization error. In this study, L-GEM is used to find the upper bound of the generalization error. According to the L-GEM, given a small positive value Q , the localized generalization error upper bound $R_{SM}^*(Q)$ of a RBFNN is given with a probability of $1-\eta$:

$$R_{SM}^*(Q) = \left(\sqrt{R_{emp}} + \sqrt{E_{S_Q}((\Delta y)^2)} + A \right)^2 + \varepsilon \quad (22)$$

where $\varepsilon = B\sqrt{\ln \eta / (-2N)}$, A , B , R_{emp} , and $E_{S_Q}((\Delta y)^2)$ denote the difference between the maximum and the minimum value of outputs, the minimum value of training mean square error, the training mean square error, and the stochastic sensitivity measure (SSM) of output differences, respectively.

Note that both A and ε are constants for a given training dataset. The SSM measures the output perturbations of a RBFNN between training samples and unseen samples in S_Q which is the union of Q -neighborhoods of all training samples. The Q -neighborhood ($S_Q(x_f)$) of a training sample x_f is defined as follows:

$$S_Q(x_f) = \{x | x = x_f + \Delta x, |\Delta x_i| \leq Q, i = 1, 2, \dots, D\} \quad (23)$$

where $\Delta x = (\Delta x_1, \dots, \Delta x_D)$, Δx_i , and D denote the stochastic perturbation, the stochastic

perturbation on the i^{th} input feature, and the number of input features, respectively. In general, we do not have any prior knowledge about the distribution of unseen samples in S_Q , thus we adopt a quasi-Monte Carlo based method (Yeung et al., 2016) to estimate the SSM value of a RBFNN $g(\cdot)$:

$$E_{S_Q} \left((\Delta y)^2 \right) = \frac{1}{N} \sum_{f=1}^N \left(\frac{1}{P} \sum_{p=1}^P \left(g(x_f + \Delta x_p) - g(x_f) \right)^2 \right) \quad (24)$$

where P is the number of Halton points and Δx_p denotes a Halton point with each coordinate ranges from $[-Q, Q]$. By fixing Q , the optimal RBFNN is found by searching the number of hidden neurons which yields the minimum $R_{SM}^*(Q)$. Note that the maximum possible number of hidden neurons is often set as the number of training samples.

3. Simulations and Results

3.1. Simulation Setup

1) Data

This study employs two 13-year (from 2005 to 2017) hourly datasets collected from the National Solar Radiation Database (Sengupta, et al., 2018) to train and test the 1-hour ahead real-life GHI forecasting models (Data from 2015 to 2016 was utilized for training forecasting models while data from 2017 was served as testing data). One dataset is based on Itupiranga (latitude = 5.15° S, longitude = 49.34° W), Brazil and the other one is based on Ocala (latitude = 29.17° N, longitude = 82.14° W), Marion, Florida, United States.

2) Implementation Details

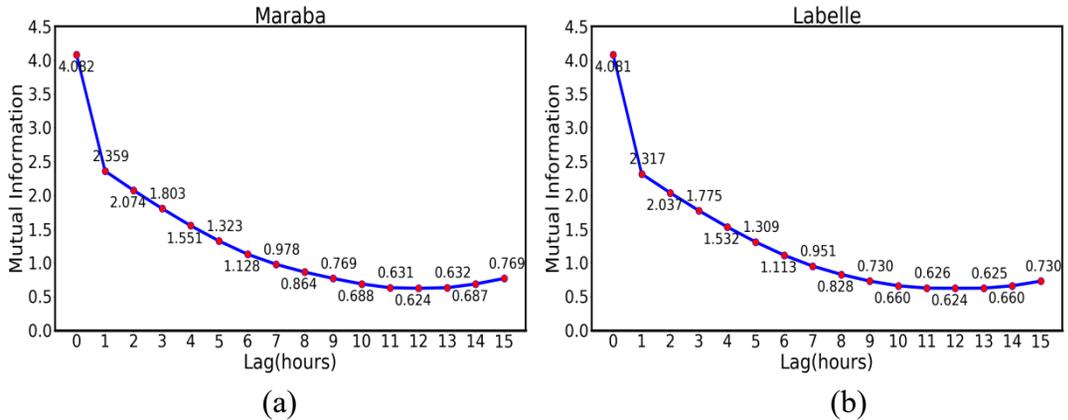


Fig. 6. Mutual information of various time-lags.

In this paper, the inputs fed to the models were the recent historical sequences pre-processed by a sliding window in this study. The size of the sliding window determines how much historical information a model can observe to forecast the next hour real-life

GHI. In this study the mutual information calculating the correlation of current observed real-life GHI with the historical real-life GHI at different time is utilized to determine the window size, as shown in Fig. 6. More specifically, the first minimum criterion (Ghimire, et al., 2019) is used to determine the window size. Hence according to Fig. 6, the sliding window size for the Maraba and the Labelle were both set as 12 hours.

For training MvDF, the training data (from year 2005 to 2016) was sorted in time to be further divided into two parts, that is, the 20% at the end of the ranking was served as the validation data for determining the hyper-parameters of the MvDF and the remaining was utilized to train MvDF. Giving the searching ranges for each hyper-parameter, those hyper-parameter configurations yielding the smallest Mean Square Error on the validation data were regarded as the best configuration. Note that the structure of each branch (i.e., TCN, BLSTMattn, and C_GRU) of MvDF was determined by the validation data separately to reduce the computational requirement. Table 1 summarizes the details of hyper-parameters configurations of MvDF. As for RBFNN, all training data from year 2005 to 2016 were utilized to train RBFNN because once the hyper-parameter Q is determined, the structure of RBFNN can be determined as well according to the minimization of L-GEM. The Q value was set as 0.01 to represent a 1% deviation from the training samples after normalization to $[0, 1]$.

Table 1. Model Structure of MvDF determined by the validation data

| Locations | Models | Hyper-parameters Configurations |
|-----------|-----------|--|
| Maraba | TCN | Two-layer causal convolutional layers; Dilation rates of the 1st and 2nd layer are 1 and 2, respectively; Kernel sizes of both layers are 2; Number of kernels of both layers is 64 ; |
| | BLSTMattn | Number of BLSTM layers is 1; Number of hidden units is 64; |
| | C_GRU | Number of convolutional layer is 1; Kernel size of the convolutional layer is 3; Number of kernel in the convolutional layer is |

| | | |
|---------|-----------|--|
| | | 64; Number of GRU layer is 1; Number of hidden units of GRU layer is 128; |
| | MvDF | The λ balancing the complementary and consensus is 0.0001; |
| Labelle | TCN | Two-layer causal convolutional layers; Dilation rates of the 1st and 2nd layer are 1 and 2, respectively; Kernel sizes of both layers are 2; Number of kernels of both layers is 64 ; |
| | BLSTMattn | Number of BLSTM layers is 1; Number of hidden units is 16; |
| | C_GRU | Number of convolutional layer is 1; Kernel size of the convolutional layer is 1; Number of kernel in the convolutional layer is 64; Number of GRU layer is 1; Number of hidden units of GRU layer is 64; |
| | MvDF | The λ balancing the complementary and consensus is 0.01; |

To show the effectiveness of the proposed method, several recent works were implemented to compare with our proposed method. The following is a brief introduction of the methods:

- (i) Smart Persistence (Yang, 2019): Smart persistence is the persistence model which is often referred to as the baseline in previous studies. The key idea is to predict the next hour real-life GHI by assuming the next hour clear-sky index is the same as the current

hour clear-sky index, as shown below:

$$\hat{y}(t+1) = k_{cs}(t)I_{cs}(t+1) \quad (25)$$

where $k_{cs}(t)$, $I_{cs}(t+1)$ and $\hat{y}(t+1)$ represent the clear-sky index at time t , the clear-sky GHI at time $t+1$, and the predicted GHI at time $t+1$.

- (ii) TB_K-means+MLP 0: A new clustering TB_K-means is proposed to partition the real-life GHI time series data into k clusters and each cluster has its corresponding real-life GHI predictor which is the Multiple Layer Perceptron (MLP).
- (iii) Random Forest (RF) (Benali et al., 2019): The random forest is a collection of multiple decision trees, where each decision tree is trained by a randomly sampled subset. The output of the RF is the average of all decision trees, which makes the RF more robust.
- (iv) LSTM (Qing and Niu, 2018): LSTM takes meteorological features as input to achieve the real-life GHI forecasting.
- (v) C_LSTM (Ghimire et al., 2019): C_LSTM exploits a CNN to extract local temporal features and then a LSTM takes these local temporal features as input to forecast real-life GHI.

Besides, to make a fair comparison, the features fed to all forecasting models (except the Smart Persistence) are the same.

3) Evaluation Criteria

Several commonly used statistical metrics for regression problems are employed to evaluate the forecasting performances of real-life GHI prediction models. They are the relative Root Mean Square Error (rRMSE), the relative Mean Bias Error (rMBE), the maximum error (Error_{\max}), and the minimum error (Error_{\min}). Equations of these metrics are given below:

$$\text{rRMSE}(\%) = \frac{\sqrt{\frac{1}{U} \sum_{t=1}^U (f(t) - h(t))^2}}{\sqrt{\frac{1}{U} \sum_{t=1}^U h(t)}} \times 100 \quad (26)$$

$$\text{rMBE}(\%) = \frac{\frac{1}{U} \sum_{t=1}^U (f(t) - h(t))}{\sqrt{\frac{1}{U} \sum_{t=1}^U h(t)}} \times 100 \quad (27)$$

$$\text{Error}_{\max} = \max |f(t) - h(t)| (\text{W/m}^2) \quad (28)$$

$$\text{Error}_{\min} = \min |f(t) - h(t)| (\text{W/m}^2) \quad (29)$$

where $f(t)$ and $h(t)$ represent the predicted GHI at time t and the real GHI at time t , respectively. U is the total number of the testing data. Furthermore, Forecast Skill (FS) is also used to make a fair comparison among point forecasts made at different locations (Yang, 2019).

$$FS(\%) = \left(1 - \frac{rRMSE_{\text{proposed}}}{rRMSE_{\text{baseline}}} \right) \times 100 \quad (30)$$

where $rRMSE_{\text{proposed}}$ is the $rRMSE$ of the model under evaluation and $rRMSE_{\text{baseline}}$ is the $rRMSE$ of the baseline (i.e., Smart Persistence).

3.2. Results and Analysis

1) Effectiveness of Multi-view Learning and Error Correction

In this study, in terms of both $rRMSE$ and $rMBE$ metrics, data from 2017 are used to evaluate the effectiveness of the proposed MvDF compared with each single network branch (view) and to evaluate the effectiveness of the error correction for MvDF. The $rRMSE$ quantifies the difference between the prediction and actual observation and the $rMBE$ quantifies how much overestimation or underestimation bias a model may produce. If a photovoltaic prediction system produces severe overestimation, it may cause the actual power shortage, otherwise, it may cause excess power generation. Therefore, a solar forecasting model which can produce the $rMBE$ as closed to 0 as possible will have a practical significance.

Table 2 shows that each branch achieves a similar performance. However, each branch provides distinct representation for real-life GHI forecasting and thus contains some information that others cannot provide. For example, the TCN makes forecasting by considering all local temporal segments, while BLSTMattn makes forecasting by utilizing the key temporal information of the working memory. When the multi-view learning schema is utilized (thus results in MvDF), the forecasting accuracies are improved (i.e., lower $rRMSE$ and $rMBE$ closer to zero). MvDF yields higher prediction accuracies than each branch and alleviates the problem of overestimation or underestimation of real-life GHI (the $rMBE$ value away from zero) on single view because it makes forecasting by jointly combining all forecasting outputs of each network branch appropriately under the complementary principle.

By utilizing the RBFNN trained via minimizing L-GEM to compensate the solar forecasting error of MvDF, MvDF_EC obtains a further improvement over MvDF especially in terms of $rMBE$. That is a significance difference of 86.5% i.e., $(0.37-0.05)/0.37$ and 33.9% i.e., $(0.59-0.39)/0.59$ for Maraba and Labelle, respectively. Although the RBFNN produces a low improvement in terms of the $rRMSE$ which may be caused by the limited capacity of the RBFNN, the RBFNN is a fast-training model compared with the deep learning models because the weight of the RBFNN can be determined by the linearly pseudo inverse method. Therefore, adding the error correction model in the forecasting

system will not complicate the system too much, but also reduce the overestimation or underestimation forecasting bias of the forecasting system.

Table 2. Results of MVDF_EC for 1-hour ahead real-life GHI forecasting

| Locations | Models | Performance Metrics | |
|-----------|-----------|---------------------|-------------|
| | | rRMSE(%) | rMBE (%) |
| Maraba | TCN | 22.57 | -1.94 |
| | BLSTMattn | 22.51 | 3.41 |
| | C_GRU | 22.37 | -0.63 |
| | MvDF | 22.10 | 0.37 |
| | MvDF_EC | 22.08 | 0.05 |
| Labelle | TCN | 27.91 | 0.98 |
| | BLSTMattn | 27.59 | -1.37 |
| | C_GRU | 27.74 | 2.71 |
| | MvDF | 27.42 | 0.59 |
| | MvDF_EC | 27.40 | 0.39 |

2) Performance Analysis

Table 3 shows the 1-hour ahead daytime real-life GHI forecasting results. The proposed MvDF_EC yields the highest FS values among all compared models, where the FS values are 15.30% and 13.29% for Maraba and Labelle, respectively. By contrast, RF achieves the lowest FS referred to the baseline (i.e., the Smart Persistence). RF makes forecasting relying on the ensemble of decision trees where a decision tree has limited prediction ability for non-stationary real-life GHI forecasting. TB_K-means+MLP achieves a higher FS than the RF because it utilizes the clustering technique to de-trend the real-life GHI time series into K clusters and develops a MLP for each cluster. However, the limited prediction ability of such shallow model (i.e., MLP) and the cluster selection strategy for testing data will affect the forecasting accuracy. By comparison, the deep learning based models (LSTM and C_LSTM) show superior performance in terms of the FS among the compared models. The proposed MvDF_EC combines the forecasting outputs of three deep learning models in

an appropriate way and utilizes an error correction model to enhance the forecasting. Thus, it obtains the highest FS values among all the forecasting models. Besides, MvDF_EC also achieves the best results in other performance metrics except the Error_{min}. MvDF_EC yields a larger Error_{min} than both Smart Persistence and RF because all deep learning based methods can only approximate the target.

Table 3. 1-hour ahead daytime forecasting performance comparisons of different models

| Location | Models | Performance Metrics | | | | |
|----------|-------------------|---------------------|-------------|------------------------------|------------------------------|--------------|
| | | rRMSE (%) | rMBE (%) | Errormax (W/m ²) | Errormin (W/m ²) | FS (%) |
| Maraba | Smart Persistence | 26.07 | -2.65 | 801.00 | 0.00 | 0.00 |
| | TB_K-means+MLP | 22.78 | 1.21 | 746.11 | 1.26E-02 | 12.62 |
| | RF | 23.87 | -0.91 | 736.00 | 0.00 | 8.44 |
| | LSTM | 22.54 | -0.62 | 783.28 | 3.88E-02 | 13.54 |
| | C_LSTM | 22.53 | -2.90 | 758.76 | 2.11E-02 | 13.58 |
| | MvDF_EC | 22.08 | 0.05 | 734.89 | 9.76E-03 | 15.30 |
| Labelle | Smart Persistence | 31.60 | 1.91 | 876.00 | 0.00 | 0.00 |
| | TB_K-means+MLP | 28.43 | 2.28 | 743.28 | 1.64E-02 | 10.03 |
| | RF | 29.56 | -0.69 | 776.80 | 0.00 | 6.46 |
| | LSTM | 27.88 | 2.76 | 765.17 | 3.36E-03 | 11.77 |
| | C_LSTM | 27.75 | 1.08 | 741.69 | 1.18E-02 | 12.18 |
| | MvDF_EC | 27.40 | 0.39 | 739.87 | 7.63E-03 | 13.29 |

To evaluate the performances of forecasting models more comprehensively, Fig. 7 shows the 1-hour ahead real-life GHI forecasting results of different models in terms of rRMSE (%) for different seasons in Maraba. The proposed MvDF_EC achieves the lowest rRMSE values compared with other models for different seasons. Besides, it yields a small rRMSE for winter but a relatively high rRMSE for summer. This is reasonable since the weather is

clear in winter while summer has a more varying and unpredictable weather.

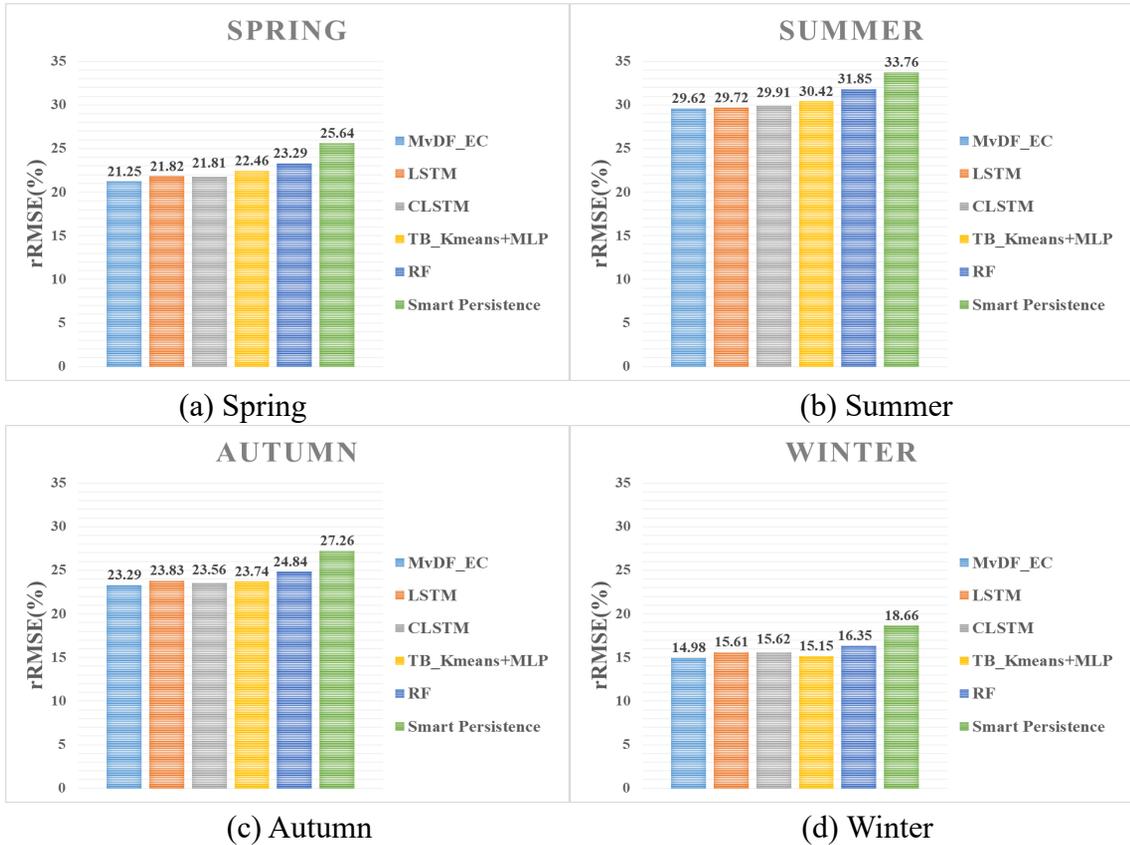
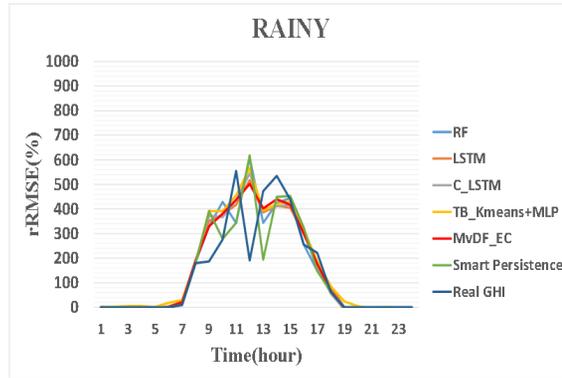
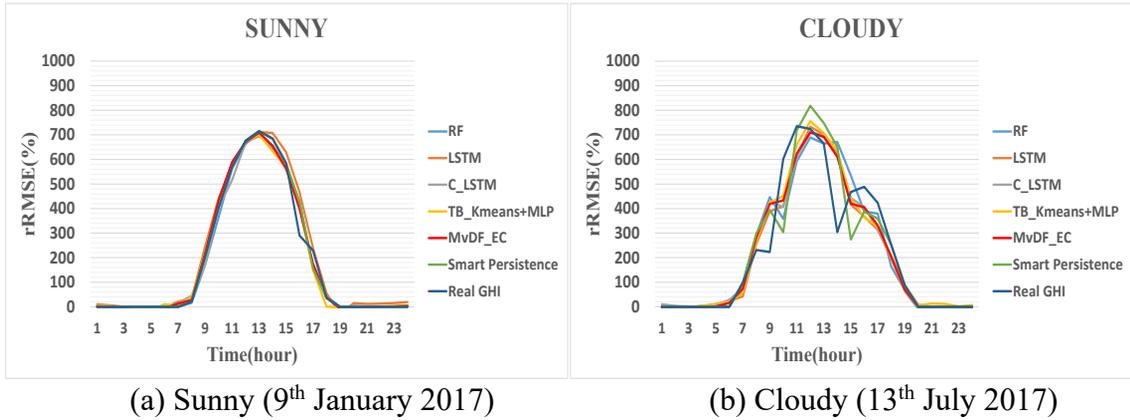


Fig. 7. 1-hour ahead real-life GHI forecasting results of different models in terms of $rRMSE(\%)$ for different seasons in Maraba.

Fig. 8 shows the real GHI series and predicted GHI series from different models under different weather condition (sunny, rainy, and cloudy) in Labelle. When it is a sunny day, all forecasting models can predict the next hour real-life GHI well. All the models perform worse when it is a rainy or cloudy day. The proposed MvDF_EC yields the smallest $rRMSE$ values with 10.18%, 31.21%, and 42.87% for the sunny, cloudy, and rainy days, respectively. MvDF_EC performs better than other models under different weather conditions because different representations from different network branches provide more information for MvDF_EC. However, the performance of MvDF_EC is still highly affected by the uncertain weather types where samples for common weather events (e.g., clear sky) dominate the training of MvDF_EC (Lai et al., 2019). Overall, the proposed MvDF_EC can give more realistic real-life GHI predictions than other solar forecasting

methods under different weather conditions.



(c) Rainy (2nd October 2017)

Fig. 8. 1-hour ahead real-life GHI forecasting results of different models for the a) sunny, b) cloudy, and c) rainy days in Labelle.

Tables 4 and 5 show the training and testing computational requirement of the proposed method and other published machine learning methods for solar forecasting on the Maraba and Labelle datasets. The experiment was conducted on the RTX 2080 Ti GPU. The training maximum epoch of all deep learning based models was set as 1000 while the early-stop technique was also used in the experiment and the patience of the early-stop technique was set as 15. As seen from Tables 4 and 5, although the training of the proposed method spends the most time in both Ocala and Itupiranga datasets compared with other machine learning based methods, the testing time of the proposed method is short enough for real-time applications. Besides, as seen from Table 1, the optimal structure of MvDF is not complicated and thus we believe the testing time of the proposed method can still meet the

real-time requirement even the GPU is not as sophisticated as RTX 2080 Ti.

Table 4. The training computational requirement (second/sample) of machine learning based models

| | TB_K-means+MLP | RF | LSTM | CLSTM | MvDF_EC |
|---------|----------------|---------|---------|---------|---------|
| Maraba | 5.82E-4 | 2.62E-3 | 3.74E-3 | 2.22E-3 | 4.60E-2 |
| Labelle | 7.13E-4 | 2.70E-3 | 5.65E-3 | 4.04E-3 | 5.10E-2 |

Table 5. The testing computational requirement (second/sample) of machine learning based models

| | TB_K-means+MLP | RF | LSTM | CLSTM | MvDF_EC |
|---------|----------------|---------|---------|---------|---------|
| Maraba | 1.48E-5 | 1.14E-6 | 2.08E-5 | 1.64E-5 | 7.19E-5 |
| Labelle | 1.74E-5 | 1.13E-6 | 1.86E-5 | 1.58E-5 | 7.65E-5 |

4. Conclusions and Future Work

This paper presents a novel Multi-view Deep Forecasting method with Error Correction (MvDF_EC) for 1-hour ahead real-life GHI forecasting. The multi-view learning schema is leveraged to jointly optimize three deep neural networks (TCN, BLSTMattn, and C_GRU) according to the consensus principle and the complementary principle, which results in the MvDF. To further improve the performance of MvDF, the robust RBFNN trained via minimizing L-GEM is adopted to correct the forecasting error of MvDF, by forecasting the prediction error of MvDF. The final solar forecasting output of the proposed MvDF_EC is the output of MvDF plus the forecasting error predicted by the RBFNN. Simulation results on the National Solar Radiation Database show that the proposed method achieves the smallest solar forecasting error compared with the smart persistence and other state-of-the-art methods. The proposed MvDF_EC yields the smallest rRMSE and the rMBE is closest to zero. For future work, we aim at incorporating MvDF_EC with de-trending techniques to take stationary data into account for achieving a lower solar forecasting error. Also, the types, structures, and the inputs of forecasting error correction model should be considered in detail.

Acknowledgements

This work is supported by National Natural Science Foundation of China under Grants 61876066 and 61572201, Guangzhou Science and Technology Plan Project 201804010245, Department of Finance and Education of Guangdong Province 2016 [202]: Key Discipline

Construction Program, China; the Education Department of Guangdong Province: New and Integrated Energy System Theory and Technology Research Group [Project Number 2016KCXTD022]; Brunel University London BRIEF Funding, UK.

References

- Azimi R., *et al.*, "A hybrid method based on a new clustering technique and multilayer perceptron neural networks for hourly solar radiation forecasting," *Energy Conversion and Management*, vol. 118, pp. 331-344, 2016.
- Bae K., *et al.*, "Hourly solar irradiance prediction based on Support Vector Machine and its error analysis," *IEEE Transactions on Power System*, vol. 32, pp. 935-945, 2017.
- Bai S., *et al.*, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *ArXiv Preprint ArXiv: 1803.01271*, 2018.
- Benali L., *et al.*, "Solar radiation forecasting using artificial neural network and random forest methods: Application to normal beam, horizontal diffuse and global components," *Renewable Energy*, vol. 132, pp. 871-884, 2019.
- Bin Y., *et al.*, "Describing video with attention-based bidirectional LSTM," *IEEE Transactions on Cybernetics*, vol. 49, no. 7, pp. 2631-2641, 2019.
- Dasgupta S., *et al.*, "PAC generalization bounds for co-training," *Advances in Neural Information Processing Systems*, pp. 375-382, 2002.
- Feng C., *et al.*, "Unsupervised clustering-based short-term solar forecasting," *IEEE Transactions on Sustainable Energy*, 2018.
- Feng C., and Zhang J., "SolarNet: A sky image-based deep convolutional neural network for intra-hour solar forecasting," *Solar Energy*, vol. 204, pp. 71-78, 2020.
- Ghimire S., *et al.*, "Deep solar radiation forecasting with convolutional neural network and long short-term memory network algorithms," *Applied Energy*, vol. 253, pp. 113541, 2019.
- Huang C., *et al.*, "Data-driven short-term solar irradiance forecasting based on information of neighboring sites," *IEEE Transactions on Industrial Electronics*, vol. 66, no.12, pp. 9918-9927, 2018.
- Han Z., *et al.*, "A review of deep learning models for time series prediction," *IEEE Sensors Journal*, 2019.
- Jiang Y., *et al.*, "Day-ahead prediction of bihourly solar radiance with a Markov switch approach," *IEEE Transactions on Sustainable Energy*, vol. 8, no. 4, pp. 1536-1547, Oct. 2017.
- Larson D.P., *et al.*, "Day-ahead forecasting of solar power output from photovoltaic plants in the American southwest," *Renewable Energy*, vol. 91, pp. 11-20, Jun. 2016.
- Lai C.S., *et al.*, "A comprehensive review on large-scale photovoltaic system with applications of electrical energy storage," *Renewable and Sustainable Energy Reviews*,

- vol. 78, pp. 439-451, 2017a.
- Lai, C.S., et al., "Daily clearness index profiles cluster analysis for photovoltaic system," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 5, pp. 2322-2332, 2017b.
- Li D., and Wang Z., "Video superresolution via motion compensation and deep residual learning," *IEEE Transactions on Computational Imaging*, vol. 3, no. 4, pp. 749-762, 2017.
- Li Y., et al., "Multi-step wind speed forecasting using EWT decomposition, LSTM principal computing, RELM subordinate computing and IEWT reconstruction," *Energy Conversion and Management*, vol. 167, pp. 203-219, 2018.
- Li C., et al., "Short-term wind speed interval prediction based on ensemble GRU model," *IEEE Transactions on Sustainable Energy*, 2019.
- Liu H., and Chen C., "Data processing strategies in wind energy forecasting models and applications: A comprehensive review," *Applied Energy*, vol. 249, pp. 392-408, 2019.
- Lai C.S., et al., "A robust correlation analysis framework for imbalanced and dichotomous data with uncertainty," *Information Sciences*, vol. 470, pp. 58-77, 2019.
- Rodríguez, F., et al., "Predicting solar energy generation through artificial neural networks using weather forecasts for microgrid control," *Renewable Energy*, vol. 126, pp. 855-864, 2018.
- Qing X., and Niu Y., "Hourly day-ahead solar irradiance prediction using weather forecasts by LSTM," *Energy*, vol. 148, pp. 461-468, 2018.
- Sanfilippo A., et al., "An adaptive multi-modeling approach to solar nowcasting," *Solar Energy*, vol. 125, pp. 77-85, 2016.
- Sengupta M., et al., "The National Solar Radiation Database (NSRDB)," *Renewable and Sustainable Energy Reviews*, vol. 89, pp: 51-60, 2018.
- Sun S., et al., "A decomposition-clustering-ensemble learning approach for solar radiation forecasting," *Solar Energy*, vol. 163, pp. 189-199, 2018.
- Su H., et al., "Adaptive residual compensation ensemble models for improving solar energy generation forecasting," *IEEE Transactions on Sustainable Energy*, vol. 11, no. 2, pp. 1103-1105, 2020.
- Wang Y., et al., "Adaptive learning hybrid model for solar intensity forecasting," *IEEE Transactions on Industrial Informatics*, vol. 14, no.4, pp. 1635-1645, 2018.
- Wu X., et al., "Optimal kernel ELM and variational mode decomposition for probabilistic PV power prediction," *Energies* 13 (14), 3592, 2020.
- Yeung D. S., et al., "Localized generalization error model and its application to architecture selection for radial basis function neural network," *IEEE Transactions on Neural Networks*, vol. 18, no.5, pp. 1294-1305, 2007.
- Yeung D. S., et al., "MLPNN training via a multiobjective optimization of training error and stochastic sensitivity," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 5, pp. 978-992, 2016.

- Yang D., "A guideline to solar forecasting research practice: Reproducible, operational, probabilistic or physically-based, ensemble, and skill (ROPES)," *Journal of Renewable and Sustainable Energy*, vol. 11, no.2, pp. 022701, 2019.
- Zhang Y., *et al.*, "Multiview convolutional neural networks for multidocument extractive summarization," *IEEE Transactions on Cybernetics*, vol. 47, no.10, pp. 3230-3242, 2016.
- Zuo Z., *et al.*, "Learning contextual dependence with convolutional hierarchical recurrent neural networks," *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 2983-2996, 2016.
- Zhao R., *et al.*, "Machine health monitoring using local feature-based gated recurrent unit networks," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 2, pp. 1539-1548, 2018.
- Zhen Z., *et al.*, "Deep learning based surface irradiance mapping model for solar PV power forecasting using sky image," *IEEE Transactions on Industry Applications*, vol. 56, pp. 3385-3369, 2020.
- Zang H., *et al.*, "Short-term global horizontal irradiance forecasting based on a hybrid CNN-LSTM model with spatiotemporal correlations," *Renewable Energy*, vol. 160, pp. 26-41, 2020.