Energy 278 (2023) 127701

Contents lists available at ScienceDirect

Energy

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

Short-term solar radiation forecasting using hybrid deep residual learning and gated LSTM recurrent network with differential covariance matrix adaptation evolution strategy

Mehdi Neshat [a,d,*], Meysam Majidi Nezhad [b], Seyedali Mirjalili [a,e], Davide Astiaso Garcia [c], Erik Dahlquist [b], Amir H. Gandomi [d,e]

[a] Center for Artificial Intelligence Research and Optimisation, Torrens University Australia, Brisbane, QLD 4006, Australia
[b] Department of Sustainable Energy Systems, Mälardalen University, Västerås, SE 72123, Sweden
[c] Department of Planning, Design, and Technology of Architecture, Sapienza University of Rome, Italy
[d] Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo, NSW, 2007, Australia
[e] University Research and Innovation Center (EKIK), Óbuda University, 1034 Budapest, Hungary

ARTICLE INFO

Keywords:
Solar radiation
Short-term forecasting
Recurrent neural network
Gated recurrent unit
Xception
Deep residual learning
Hybrid deep learning models

ABSTRACT

Developing an accurate and robust prediction of long-term average global solar irradiation plays a crucial role in industries such as renewable energy, agribusiness, and hydrology. However, forecasting solar radiation with a high level of precision is historically challenging due to the nature of this source of energy. Challenges may be due to the location constraints, stochastic atmospheric parameters, and discrete sequential data. This paper reports on a new hybrid deep residual learning and gated long short-term memory recurrent network boosted by a differential covariance matrix adaptation evolution strategy (ADCMA) to forecast solar radiation one hour-ahead. The efficiency of the proposed hybrid model was enriched using an adaptive multivariate empirical mode decomposition (MEMD) algorithm and 1+1EA-Nelder–Mead simplex search algorithm. To compare the performance of the hybrid model to previous models, a comprehensive comparative deep learning framework was developed consisting of five modern machine learning algorithms, three stacked recurrent neural networks, 13 hybrid convolutional (CNN) recurrent deep learning models, and five evolutionary CNN recurrent models. The developed forecasting model was trained and validated using real meteorological and Shortwave Radiation (SRAD1) data from an installed offshore buoy station located in Lake Michigan, Chicago, United States, supported by the National Data Buoy Centre (NDBC). As a part of pre-processing, we applied an autoencoder to detect the outliers in improving the accuracy of solar radiation prediction. The experimental results demonstrate that, firstly, the hybrid deep residual learning model performed best compared with other machine learning and hybrid deep learning methods. Secondly, a cooperative architecture of gated recurrent units (GRU) and long short-term memory (LSTM) recurrent models can enhance the performance of Xception and ResNet. Finally, using an effective evolutionary hyper-parameters tuner (ADCMA) reinforces the prediction accuracy of solar radiation.

1. Introduction

Solar power is one of the most abundant, accessible, and infinitely renewable energy sources yielded when energy from sunlight is transformed into electricity. Solar power is considered a top alternative source to fossil fuels with a high potential to meet global energy demands in the near future [1]. In solar energy technologies, the development of an accurate prediction short-term or long-term (day ahead) [2] solar radiation model plays a fundamental role in enhancing the scheduling and controlling the performance of photovoltaic power plants; having a reliable and robust plan for managing connection to smart grids [3]; and improving the gain margin of the energy suppliers in these markets.

However, predicting solar energy is challenging due to solar radiation's intermittent and chaotic nature and atmospheric situations that are naturally ungovernable (i.e., clouds, shadows, the vapour of water, ice, air pollution or aerosols in the atmosphere) [4]. Another

* Corresponding author at: Center for Artificial Intelligence Research and Optimisation, Torrens University Australia, Brisbane, QLD 4006, Australia.
E-mail addresses: mehdi.neshat@torrens.edu.au (M. Neshat), meysam.majidi.nezhad@mdu.se (M.M. Nezhad), ali.mirjalili@torrens.edu.au (S. Mirjalili), davide.astiasogarcia@uniroma1.it (D.A. Garcia), Erik.Dahlquist@mdu.se (E. Dahlquist), gandomi@uts.edu.au (A.H. Gandomi).

https://doi.org/10.1016/j.energy.2023.127701
Received 19 January 2023; Received in revised form 26 April 2023; Accepted 29 April 2023
Available online 10 May 2023

Table 1
All the acronyms utilised and arranged in a sequential order based on the alphabet.

Abbreviation	Full name
ADCMA	Differential covariance matrix adaptation evolution strategy
AOA	Arithmetic optimisation algorithm
AR	Auto-regressive models
ARIMA	Auto-regressive integrated moving average models
AI	Artificial intelligence
ANFIS	Adaptive neuro-fuzzy inference system
ANN	Artificial Neural networks
BPNN	Back-propagation Neural Network
Bi-LSTM	Bidirectional Long short-term memory network
BS	Batch size
CART	Classification and regression tree
CEEMDAN	Adaptive ensemble decomposition method
CR	Probability crossover rate
CMA-ES	Covariance matrix adaptation evolution strategy
CNN	Convolutional neural network
CS	Cuckoo Search
DBN	Deep belief network
DE	Differential evolution
DNN	Deep neural networks
EEMD	Ensemble empirical mode decomposition
ELM	Extreme Learning Machine
EMD	Empirical mode decomposition
ER	Energy ratio
FFNN	Feed-forward neural networks
GA	Genetic algorithm
GBT	Gradient boosting tree
GRU	Gated recurrent unit
GNDO	Generalised normal distribution optimisation
IMF	Intrinsic mode functions
LOO-CV	Leave-one-out cross-validation
LSTM	Long short-term memory network
MAE	Mean absolute error
MARS	Multivariate adaptive regression spline
MEMD	Multivariate empirical mode decomposition
ML	Machine learning
MLP	Multi-layer perceptron
MOA	Mathematical Optimiser Accelerated
MSE	Mean square error
MSR	Multi-response Sparse Regression
NDBC	National Data Buoy Centre
NM	Nelder–Mead simplex direct search method
PNN	Polynomial neural networks
RMSE	Root mean square error
RNN	Recurrent neural networks
SCA	Sine cosine meta-heuristic algorithm
SGD	Stochastic gradient descent
SRAD	Shortwave Radiation
SRT	Sifting relative tolerance
SMAPE	Symmetric mean absolute percentage error
SVM	Support vector machines
VMD	Variational mode decomposition

primary motivation for predicting solar radiation is that installing and maintaining solar radiation measurement devices is highly costly. This makes installing such instruments in every meteorological station financially challenging, especially in developing countries. As an example, there were around 1800 meteorological stations in Turkey in 2020; however, just 7% of them were equipped to register solar radiation data [5]. From this perspective, various technical models have been proposed to forecast solar radiation. The empirical model is one of the popular prediction models established on mathematical procedures. Its benefits include fast and straightforward calculations and is helpful techniques for predicting long-term (monthly or weekly) solar radiation data [6]. However, empirical models cannot accurately predict short-term solar radiation data due to changeable parameters in weather conditions such as cloud cover, rainy days, etc. Furthermore, extracting the intricate and nonlinear associations found in the dependent and independent variables is challenging for empirical models, particularly in humid subtropical climate areas when the weather is rainy with heavy clouds cover [7].

Previous solar energy research studies have proposed considerable number of data-driven techniques for short-term and long-term forecasting. These techniques fall into three main areas, physical techniques, statistical analysis, and machine-deep learning methods. In the physical forecasting models, the atmosphere's dynamic motion and physical conditions are characterised using a set of mathematical formulas. The performance of physical models relies heavily on the quality and quantity of meteorological variables and astronomical dates (e.g., solar time and earth declination angle) [8]. The statistical approaches using statistical analysis of the various intake features for solar radiation prediction have been applied, including the auto-regressive models (AR), auto-regressive integrated moving average (ARIMA) [9], exponential smoothing, Markov Chain model [10] and Gaussian process [11]. Most of them show acceptable accuracy for predicting the ground solar radiation and cloud motion on different time horizons up to hours ahead. In the last decades, the application of artificial intelligence (AI)-based approaches has considerably developed in solar engineering fields [12]. Previous analyses represent that the AI-based approaches are able to provide more accurate forecasting of solar radiation results than those of the other models [13] such as supervised and unsupervised artificial neural networks (ANN) [14], deep learning models [15], support vector machines (SVM) [16], etc.

A comparative study [17] was done to clarify which one of the six machine learning models can perform best, including the gradient boosting tree (GBT), multi-layer perceptron (MLP), standard ANFIS, subtractive and fuzzy c-means clustering ANFIS, classification and regression tree (CART), and multivariate adaptive regression spline (MARS) to forecast solar irradiation in two sites. The Ref. [17] recommended applying the GBT model as a robust and reliable tool for predicting solar radiation.

A popular sequential deep learning model called long short-term memory (LSTM) is one of the most successful tools in handling the dependency between successive time series data with short-term intervals. One considerable early study was done by Qing and Niu [18] investigated hourly solar radiation forecasting using LSTM. The proposed LSTM [18] was %18 more precise than BPNN in RMSE.

In short-term solar radiation forecasting, recurrent neural networks (RNN) promise high accuracy and robustness in relation to, long short-term memory (LSTM), bidirectional LSTM, and Gated Recurrent Units (GRU). However, initialising the hyper-parameters of RNNs is challenging due to the complex and non-linear relationships between the setting parameters and the topology and nature of the time-series data. To address these problems, Peng et al. [19] developed a hybrid deep learning model combination of BiLSTM, an adaptive ensemble decomposition method (CEEMDAN), and a sine cosine meta-heuristic algorithm (SCA) for predicting hourly stochastic historical time series solar radiation data. The comparative modelling results suggested that the proposed hybrid model [19] could conquer seven other machine learning models.

Nevertheless, increasing the time horizon for forecasting solar radiation is challenging [20] for AI-based methods because of decreased auto-correlation among the time series samples. One preliminary study in long-term global solar radiation by Jiang [21] applied traditional neural networks (feed-forward back-propagation) and compared them with different empirical regression methods. The findings [21] confirmed the superiority and high ability in generalising ANN models demonstrate in solar radiation forecasting. Multilayer perception (MLP) is one of the most popular and classic machine-learning techniques and has been applied in several studies in forecasting solar radiance [22]. In early work [23], Rodriguez et al. applied a combination of five Multilayer perceptron feed-forward neural networks (an ensemble model) developed by a Monte Carlo simulation to forecast global solar radiation, and the overall validation error and accuracy were considerable. However, the drawbacks of fully connected networks (e.g. network overfitting) were not considered in [23]. To deal with the long-term solar forecasting challenges, Kisi [24] analysed and compared the three

Table 2
A list of all mathematical symbols applied in this study.

Symbols	Description	Symbols	Description
I_M	Multivariate inputs	$r_i(t)$	residual for each variate
V_i^{β}	vectors of direction	β^c	angles
T	number of directions	$env_{mean}(t)$	multidimensional envelops
$\mathcal{N}$	normal distribution	μ_i	mean
σ_i^2	standard deviation	$f(\pi)$	frequency of the signal
$p(\pi)$	relative frequency	$H_p(n)$	permutation entropy
N_v	number of parameters	Z_t	update gate
w_Z	update gate weights	x_t	hidden layer input
h_{t-1}	output of the final hidden layer	rt	reset gate
w_r	reset gate weights	$tanh$	hyperbolic-tangent
$\vec{V}_g$	differential vector	$\vec{A}_r$	random selected solutions
f	mutation factor	CR	probability crossover rate
U_g	trial vector	C_g	covariance matrix of the solutions
P_f	penalty factor	$\theta_1,\ \theta_2$	random variables
M_l	current population's position average	$s_1,\ s_2,\ s_3$	three solutions randomly
N_{pop}	population size	d	problem dimension
I_{it}	current replication number	Max_{it}	maximum number of evaluations
γ	acceleration coefficient	ξ	sensitivity factor
C_o	control parameter	B	orthonormal basis of eigenvectors
T', T	predicted and target sample	α_1, α_2	random numbers [0-1]

AI-based methods: fuzzy genetic, ANN, and neuro-fuzzy models for estimating monthly solar radiations from the Mediterranean areas. The modelling results indicated that the fuzzy genetic model could perform better than the other two models. In another study [25] examining monthly solar radiation prediction, a traditional ANN and adaptive neuro-fuzzy inference system (ANFIS) were applied. The prediction results illustrated that ANFIS mostly outperformed other ANNs. However, the study did not discuss the importance of hyper-parameters initialisation [25].

One initial effort to apply deep learning models such as convolutional neural networks (CNN), was proposed by Kaba et al. [26] in order to estimate daily global solar radiation. Although the technical details of the CNN model are not clear, the estimation results show that the CNN model can be an appropriate alternative approach in long-term solar radiation forecasting.

To tackle the issues raised by traditional neural networks training, such as the exhaustive learning process, insufficient parameter preference, and the need for a large number of training samples, a new model of neural network called a Deep Belief Networks (DBN) [27] was proposed in 2007. A combined DBN [28] with a clustering idea was used to develop an accurate daily solar energy forecasting model based on 30 sites located in China. The DBN method [28] acquired more reasonable accuracy from the outcomes than empirical ML methods. Tuning the hyper-parameters of deep learning models is essential; nevertheless, it is frequently challenging. Meta-heuristic algorithms have been applied in order to optimise the hyper-parameters that lead to improving the average performance of the models. Wang et al. [29] proposed a primary hybrid solar radiation forecasting models consisting of an Extreme Learning Machine (ELM) and Cuckoo Search (CS). In order to reduce the computational runtime, the Ref. [29] applied a combination of Multiresponse Sparse Regression (MSR) and leave-one-out cross-validation (LOO-CV) to determine the priority of neurons and remove the lowest ones in Feed Forward Neural Networks. The CS played the role of weight coefficients optimiser.

Despite comprehensive studies in the last years, at least three research gaps remain in designing techniques/models for short- and long-term solar radiation prediction as follows:

1. One of the most significant factors in improving the performance of prediction models hyper-parameters tuning.
2. In most case studies, optimising the architecture of deep learning models using various recurrent neural networks did not consider substantially.
3. The low performance of solar radiation predictors is due to an insufficient decomposition setting .

This study proposes a novel hybrid residual deep learning model for forecasting solar radiation one hour ahead based on real meteorological and Shortwave Radiation (SRAD1) data from an installed offshore buoy station located in Lake Michigan, Chicago, United States and supported by the National Data Buoy Centre (NDBC). In order to clean the data and improve accuracy, an outlier detection method (autoencoder) was applied. A hyper-parameter optimiser is also proposed to reinforce the model's performance. The foremost contributions of this study are summed as follows:

1. A novel hybrid solar radiation forecasting model is proposed composed of recurrent neural networks (GRU, LSTM and BiLSTM), and a convolutional ResNet50 model (deep residual learning) with adaptive decomposition technique and effective autotuner (ADCMA-ResNet50-GRU-2LSTM).
2. An adaptive multivariate empirical mode decomposition (MEMD) algorithm is proposed to decompose solar radiation time-series data with a high level of nonlinearity and non-stationarity into intrinsic mode functions (IMFs) with minimum entropy using an evolutionary Nelder–Mead simplex search algorithm.
3. In order to deal with the shortcomings of the hyper-parameters tuning initialisation, an effective and smart hyper-parameters tuner, adaptive differential covariance matrix evolutionary algorithm (ADCMA), was developed to improve prediction accuracy and reduce modelling bias.
4. A comprehensive comparative framework was also designed to evaluate the performance of various canonical and hybrid machine learning and deep learning models with regard to developing an accurate and reliable solar radiation forecasting model.

The principle sections of this article are organised as follows. The technical details of the involved methods are exemplified in the next Section 2. The following presents the case study's attributes and their statistical analysis of the dataset used in this study in Section 3. In order to develop a systematic comparison framework for the short-term solar radiation forecast, various models are evaluated and compared with the proposed model in Section 4. Eventually, in Section 5, the acquired results of this investigation and future research plans are outlined. All the acronyms and symbols utilised in this study can be seen in Tables 1 and 2.

2. Materials and methods

2.1. Time-domain signal decomposition

Several decomposition techniques can be used to extract the primary characteristics of complex and nonlinear time-series data. They involve a robust statistical approach that disintegrates a nonlinear signal down into some elements based on a directional, periodical and stochastic element. The different applications of these features can forecast, predict or infer unseen data [30]. The most popular time-series decomposition methods are variational mode decomposition (VMD) [31], multivariate VMD [32], Fourier's analysis [33], wavelet analysis [34] and empirical mode decomposition (EMD) [35]. Studies [36] have demonstrated that Fourier's analysis performance needs to be modified to extract the feature of time-series data. Furthermore, if the time-series data is not continuous, the wavelet analysis cannot be a proper selection for data feature extraction [37]. The EMD procedure is able to drag attributes for continuous and discrete transformations. In addition, EMD shows robust performance in extracting attributes from non-linear time-series data.

2.1.1. Multivariate empirical mode decomposition (MEMD) algorithm

Multivariate EMD [38] is a popular modified version of EMD used to handle multi-channel data analysis and decompose time-series data with a high level of nonlinearity and non-stationarity into IMFs. Each IMF includes a specific frequency that shows a level of decomposition, and the longest wavelength is associated with the initial IMF. The signal frequency substantially declines by increasing the number of IMF, and the decomposition process's final element denotes residual values. Despite the advantages of the EMD method in decomposing complex time-series data into IMFs and residuals, this method has some fundamental drawbacks for decomposing multi-dimensional time-series data. First, decomposing variables, which are time series using EMD, may not lead to similar frequency for IMFs. Moreover, due to the internal adaptive extraction technique of EMD, the number of IMFs (components) may differ for each input variable (time series) [39]. In order to improve EMD's weak points and reduce the computational cost, a multivariate EMD (MEMD) approach is proposed [38].

In MEMD, multivariate inputs $I = \{I_1(t), I_2(t), \dots, I_M(t)\}$ can be evaluated. The IMFs are extracted simultaneously, where each time series variable decomposes into some IMFs and residual values ($I_{(i=1:M)} = \{imf_i^1(t), imf_i^2(t), \dots, imf_M^K(t) + r_i(t)\}$), M and K show the number of variables and IMFs, and $r_i(t)$ denotes a residual for each variate. The MEMD algorithm is described as follows.

1. After receiving the raw multi-channel data $I = \{I_1(t), I_2(t), \dots, I_M(t)\}$, the Hammersley function is used to generate the initial population of time-series samples made by input signals.
2. The extracted angles from standardised Hammersley sequences are re-scaled between 0 and 2π. Then the vectors of direction define $\vec{V}^{\beta_i} = \{\vec{v}_1^c, \vec{v}_2^c, \dots, \vec{v}_M^c\}$, $c = 1, 2, \dots, T$, and these vectors are associated with angles $\beta^c = \{\beta_1^c, \beta_2^c, \dots, \beta_{m-1}^c\}$, where T is the number of directions. Using this angle of cth direction, the projection of I is calculated $E^{\beta}(t)_{c=1}^{T}$.
3. Next, the projection extreme is computed based on the instantaneous moment. Then, the coordination of the extreme points will be calculated.
4. In order to achieve the enveloped curves of I, the spline interpolation is used. Finally, the local average of the multidimensional envelops is computed using $env_{mean}(t) = \frac{1}{(T-N_e)} \sum_{c=1}^{T-N_e} e_c^{\beta}(t)$.
5. using a repetitive process of sifting, the multidimensional signal can be decomposed into $I(t) = \sum_{i=1}^{M} imf_i(t) + r_K(t)$

2.1.2. Adaptive MEMD using 1+1EA-Nelder–Mead simplex search algorithm

Although the performance of MEMD is higher than standard EMD's efficiency in terms of the decomposition of non-linear signals and computational cost, tuning the control parameters of MEMD is challenging because of existing nonlinear relationships. This study focuses on tuning six parameters of MEMD, the number of IMFs, the interpolation method for envelope construction ('spline' and 'pchip'), the maximum energy ratio (MER), which is an energy coefficient of the target signal in the first step of sifting and the mean energy of envelope, the maximum number of -maxima or -minima values (MNE) corresponding to the residual signal, sifting iterations (maximum number), and sifting relative tolerance (SRT) [40], which is a stop criterion of a Cauchy-type (Eq. (1)).

$$SRT \simeq \frac{\| r_{i-1}(t) - r_i(t) \|_2^2}{\| r_{i-1}(t) \|_2^2} \tag{1}$$

where r_i and r_{i-1} are the current and previous residuals of the original signal. Also, the equation below is used to find the current residual signal's energy ratio (ER).

$$ER \simeq 10 \log_{10}\left(\frac{\| X(t) \|_2}{\| r_i(t) \|_2}\right) \tag{2}$$

To find the optimal MEMD parameters, we proposed a fast and effective adaptive search algorithm which is a combination of 1+1EA with an adaptive mutation step size and a Nelder–Mead (NM) simplex search algorithm. The search process starts with a random solution of six parameters based on a normal distribution with $\mathcal{N}(\mu_i, +\sigma_i^2)$, $\mu = abs(UB - LB)/2$ and $\sigma = abs(UB - LB)/4$, where UB and LB are an upper bound and lower bound of the parameters.

We apply the permutation entropy method [41] to evaluate the fitness of each solution which is the output of MEMD. If $f(\pi)$ is associated with the frequency of the signal, then we can compute the relative frequency by $p(\pi) = \frac{f(\pi)}{T-(n-1)l}$, and the permutation entropy is formulated as follows: $H_p(n) = -\sum_{n=1}^{n!} p(\pi) ln p(\pi)$. The $H_p(n)$ value lower than one shows the time series data with lower complexity.

This adaptive search algorithm starts with a popular single-based solution evolutionary algorithm 1+1EA [42], with a standard mutation strategy to generate a new solution with mutation probability $\frac{1}{N_v}$ where N_v is the MEMD parameter. The σ value is set based on the search progress. If the performance of 1+1EA improves, the σ decreases linearly. In contrast, by raising the size of σ, the exploration ability of the search method develops to provide more chances of escaping a local optimum. The 1+1EA is changed by the Nelder–Mead algorithm after a few iterations, and these two methods cooperate to optimise the average entropy of all IMFs using an alternative strategy. Fig. 1 demonstrates the extracted ten IMFs plus residual components of four features, SRAD, wind speed, wind direction and air temperature. Furthermore, as can be seen in Fig. 1, the proposed adaptive decomposition method can successfully decompose nonlinear and non-stationary signals into IMFs, which can capture the nonlinear and non-stationary features of the signal. In contrast, linear methods like Fourier analysis and wavelet analysis assume that the signal is stationary and linear, which limits their applicability to non-stationary and nonlinear signals.

Fig. 2 shows the performance of the proposed hybrid optimisation method to minimise the average entropy of all IMFs. We can see that all ten independent optimiser tests converged around 0.9 with a few evaluations. From Fig. 2, the proposed 1+1EA-Nelder–Mead simplex search algorithm is a robust optimisation algorithm that combines the strengths of both the 1+1EA and Nelder–Mead methods. It is efficient, powerful, has low memory requirements, is easy to implement, can be easily parallelised, adapts its search step size to the function landscape, and does not require a population size. These features make it a valuable tool for many optimisation problems, including those with large solution spaces, noisy and non-smooth functions, and those requiring fast and efficient optimisation.

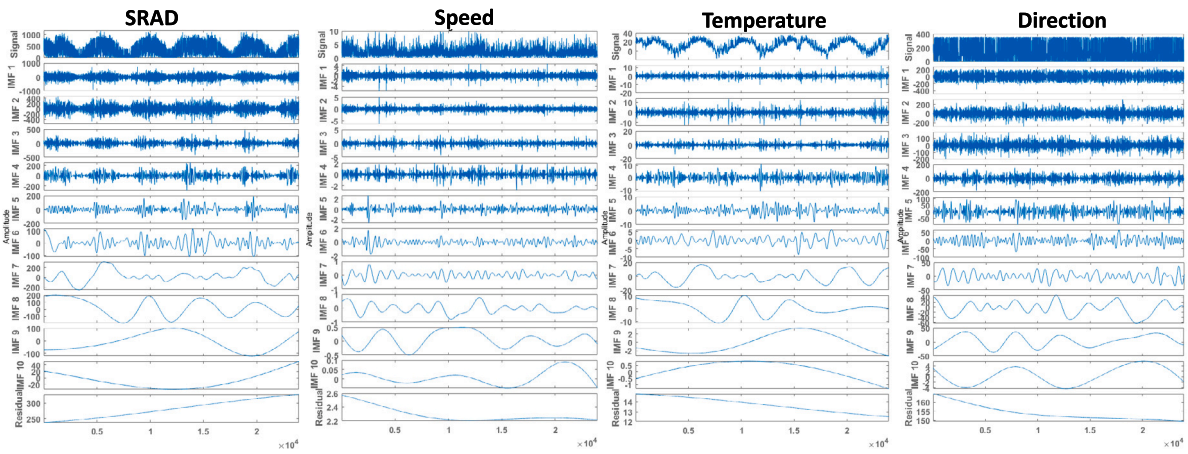

Fig. 1. The decomposition results of g Adaptive MEMD for four features: SRAD, wind speed, air temperature, and wind direction.

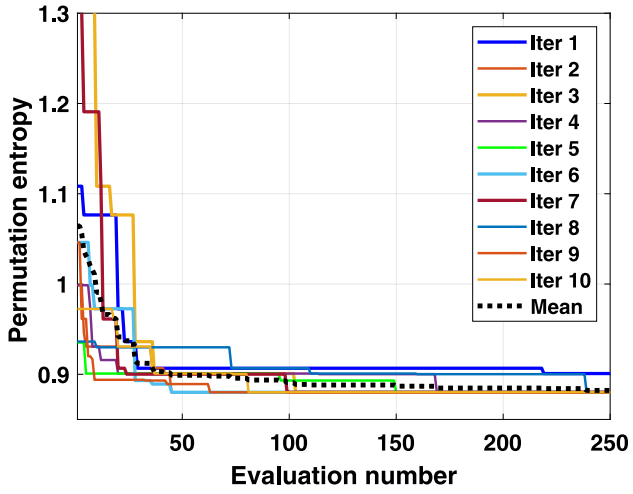

Fig. 2. The convergence rate of ten independent runs of adaptive MEMD using 1+1EA-Nelder–Mead simplex search algorithm.

The decomposition hyper-parameters trajectory of the 1+1EA-NM algorithm based on the best iteration can be seen in Fig. 3. The best-found hyper-parameters of the MEMD are IMFs' number=10, Interpretation method='spline', Maximum energy ratio=65, Extrema number=1, Sift maximum iteration=87, and sift relative tolerance = 0.49.

2.2. Gated recurrent unit

One of the most critical problems in developing an RNN is vanishing or exploding the gradients in the time-series dataset. The main objective of proposing the GRU [43] is to deal with this issue. GRU models are able to successfully extract the short-term and long-term inter-relationship dependencies among time series data. The primary structure of a GRU model includes three layers, an input layer, a hidden layer (an update gate and a reset gate) and an output layer [44]. Eq. (3) shows how the update gate (Z_t) is computed.

$$Z_t = \sigma(\omega_Z.[h_{t-1}, x_t]) \tag{3}$$

where the activation function ($sigmoid$) and update gate weights show by σ and ω_Z, respectively. x_t and h_{t-1} are the hidden layer input and the output of the final hidden layer. The role of the update gate is to determine the domain size of previous data samples, which should be involved in forecasting the next situations. Eq. (4) shows the update calculation of the reset gate (rt).

$$r_t = \sigma(\omega_r.[h_{t-1}, x_t]) \tag{4}$$

where w_r is the reset gate weights. The reset gate defines what period of the previous steps should be eliminated. Therefore, the current memory (h'_t) unit should be updated as follows:

$$h'_t = tanh(\omega.[r_t.h_{t-1}, x_t]) \tag{5}$$

where $tanh$ denotes the activation function. The last memory cell at the current time episode (h_t) is achieved as follows.

$$h_t = (1 - Z_t)h_{t-1} + Z_t h'_t \tag{6}$$

where Z_t, h_{t-1} and h_t are the update gate output, hidden layer output, and the current memory content, respectively.

2.3. Xception: Deep learning with depth-wise separable convolutions

The Xception deep learning architecture stands as an unbent heap of depth-wise separable convolution layers with residual ties to efficiently specify and adjust the deep grid architecture and focuses on "extreme inception" [45]. Xception is a well-known modified version of the deep Inception model that substitutes traditional inception modules with different depth convolutions [46]. The prominent Xception hypothesis is that in convolutional feature maps, mapping of both spatial and cross-channel correlations should be fully disengaged. The standard architecture of Xception includes three sections: Entry, Middle and Exit. The Xception convolutional layers number is 36 (the total number of layers = 170 with 22.8 million parameters) that make up the core of feature extraction and merge into 14 modules, all of which, except for the initial and final modules, are connected to linear residual connections. In the entry flow, we can see (Fig. 4) four convolutional modules stem from two convolutional layers with a rectified linear (ReLU) activation function. In the following, three modules in the entry flow have the same architecture of Separable Convolution layers mixed with ReLU and MaxPooling. There are eight modules of Separable Convolution layers with ReLU in the middle flow of Xception. Finally, in the exit flow, we have two sequential modules consisting of three Separable Convolution layers mixed with ReLU and MaxPooling. Depending on the application of Xception, the flattened, fully connected, and logistic regression layer can be arrayed at the final step. According to the reference, the hyper-parameters and configurations are as follows: On ImageNet: initial learning rate= 0.045, Learning Rate Drop Factor = 0.94, Learning Rate Drop Period = 2, Solver = SGD, and Momentum (Contribution of the previous step) = 0.9.

2.4. ResNet50: Deep residual learning

One of the convolutional deep learning models successful in a wide range of classification and regression problems is ResNet [47]. The

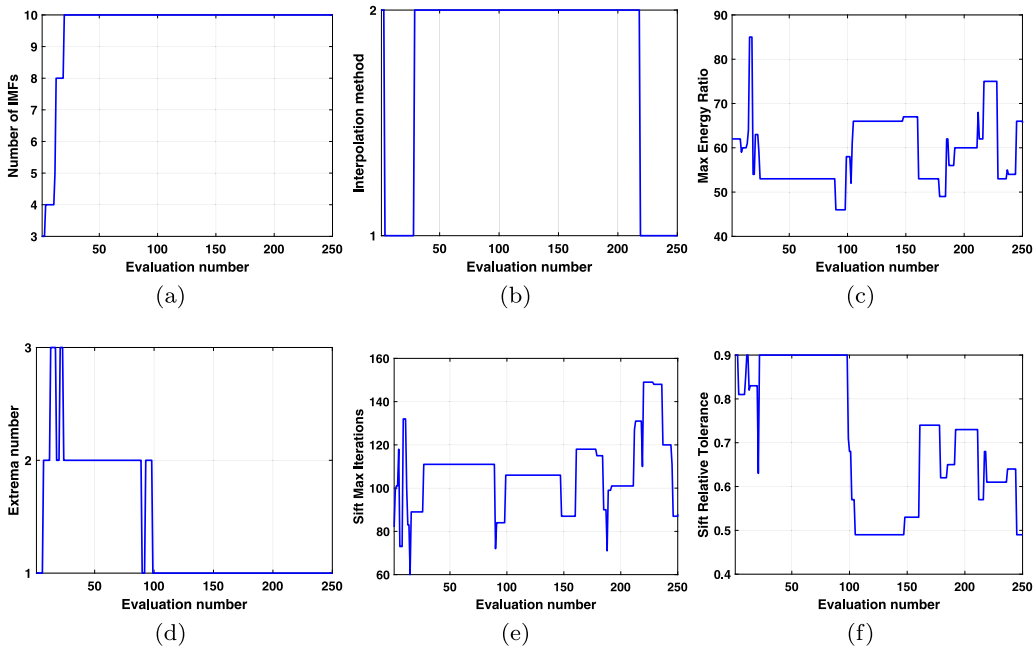

Fig. 3. The trajectory of best-found hyper-parameters using adaptive MEMD. (a) the number of IMFs, (b) the method of interpretation, (c) the Maximum energy rate, (d) the number of extrema, (e) the maximum number of sift, (f) the relative tolerance of sift.

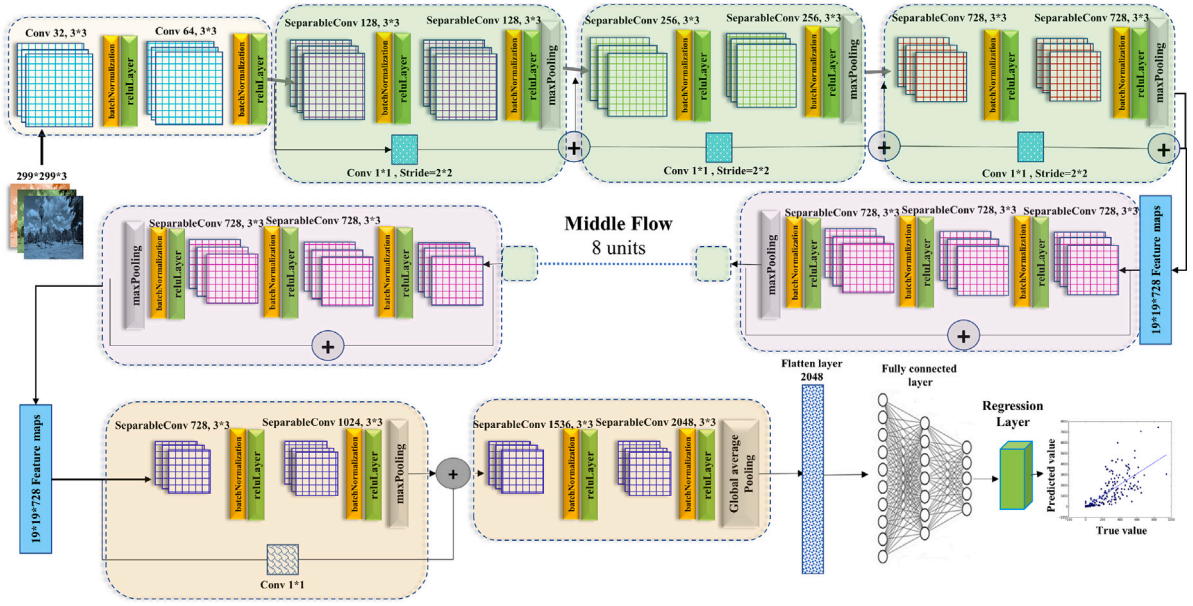

Fig. 4. A schematic of the Xception deep learning model: initially, the data proceeds via the entry flow (light green modules), Next through the middle flow that includes eight modules, and ultimately through the exit flow. Whole Separable Convolution layers utilise a depth multiplier of 1 (without depth elaboration).

general concept of modern deep learning methods is that increasing the convolutional layers improves the average performance of the deep model. However, training the deep neural networks can be problematic. One issue is the disappearance of the gradient due to its back-propagating to previous layers. In addition, iterated accumulation may cause the gradient to become infinitesimally miniature. After introducing ResNet by Kaiming et al. [47] in 2016 and proposing a shortcut connection to adjust the prior layer output to the input of the next layer without any change, this hypothesis (an intense model) was rejected. A schematic of the ResNet50 architecture can be seen in Fig. 5.

The shortcut modules applied in the ResNet (observed in Fig. 5) include (1) an Identity block: which has no convolution layer at the shortcut and whose input flow dimensions are the same as the output, (2) a Convolutional block: which consists of a convolution layer with batch normalisation at the shortcut. The dimension of the input reduces the convergence rate. As seen in Fig. 5, there is a 1×1 convolutional layer connected first and final units of the model. This strategy is named 'bottleneck' and leads to shortening the models' training parameters with considering the model's performance. For both models, when the shortcuts proceed over feature maps of two sizes, a stride of 2 is used. Overall, ResNet50 has several benefits [48] compared to other convolutional models, including a deeper architecture, residual connections, improved accuracy, transfer learning capabilities, and robustness to noisy data. These advantages make ResNet50 valuable in the deep learning toolbox for image classification tasks and other applications.

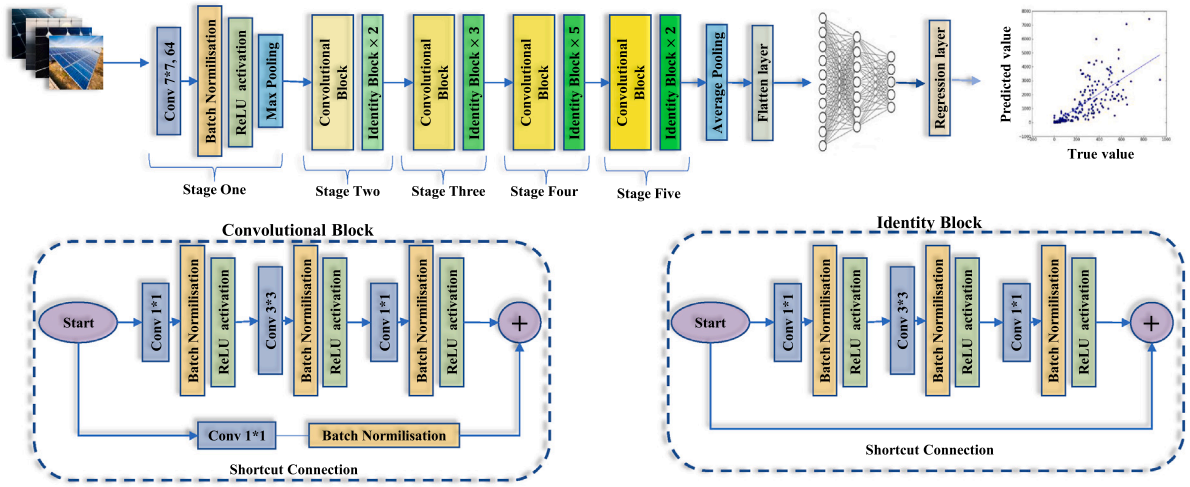

Fig. 5. A schematic of the ResNet50 deep residual learning model.

2.5. Bio-inspired hyper-parameters tuners

2.5.1. Differential evolution (DE)

Storn and Price in [49] introduced the application of differential vectors in the form of a triangle search as a new evolutionary algorithm called differential evolution (DE). DE is one of the most popular population-based optimisation methods used in optimising a wide range of noisy, dynamic, and multi-modal real engineering problems [50]. The operator of the mutation can be recognised as the most significant function in DE. In the following, some of the popular DE mutation operators propose, and their convergence rate and exploration abilities describe.

The "DE/rand/1/bin" strategy usually illustrates a low convergence rate and intuitively supports the stronger exploration ability. Consequently, it can usually be more proper for figuring out multi-modal optimisation problems than the strategies making known the best candidate chosen so far.

$$\overrightarrow{V}_g = \overrightarrow{A}_{r1,g} + f \times (\overrightarrow{A}_{r2,g} - \overrightarrow{A}_{r3,g}) \tag{7}$$

where $\overrightarrow{V}_g$ is differential vector of three random selected solutions, $\overrightarrow{A}_{r1,g}$, $\overrightarrow{A}_{r2,g}$, and $\overrightarrow{A}_{r3,g}$. Meantime, f is the mutation factor related to adjusting the exploration step size.

The second popular mutation method is DE/best/1/bin, which is generally able to provide a fast convergence speed, especially for unimodal problems. Nevertheless, it is no more likely to handle a local optimum. Thus, this strategy leads to premature convergence when searching multi-modal problems. This mutation can be formulated using Eq. (8).

$$DE/best/1/bin : \overrightarrow{V}_{i,g} = \overrightarrow{A}_{best,g} + f.(\overrightarrow{A}_{r1,g} - \overrightarrow{A}_{r2,g}) \tag{8}$$

This strategy ("DE/current-to-best/1/bin") is practically more robust than the "DE/best/1/bin". This is because it employs a consequence of two differential vectors. Therefore, both prohibiting a premature convergence rate and a moderate convergence speed will be acquired.

$$DE/current-to-best/1/bin : \overrightarrow{V}_{i,g} = \overrightarrow{A}_{i,g} + f.(\overrightarrow{A}_{best,g} - \overrightarrow{A}_{i,g}) + f.(\overrightarrow{A}_{r1,g} - \overrightarrow{A}_{r2,g}) \tag{9}$$

Empirically, strategies based on two difference vectors may develop a better perturbation than one differential trail. This was demonstrated by Storn [51]; according to the central limit theorem, when a strategy based on two difference vectors is applied, its statistical distribution is a bell curve, which is regarded as a much better perturbation. To sum up, "DE/best/2/bin" is equipped with an effective exploration technique.

$$DE/best/2/bin : \overrightarrow{V}_{i,g} = \overrightarrow{A}_{best,g} + f.(\overrightarrow{A}_{r1,g} - \overrightarrow{A}_{r2,g}) + f.(\overrightarrow{A}_{r3,g} - \overrightarrow{A}_{r4,g}) \tag{10}$$

Moreover, The "DE/rand/2/bin" strategy proposes a powerful exploration capability using Gaussian-like perturbation and an average of five randomly selected solutions.

$$DE/rand/2/bin : \overrightarrow{V}_{i,g} = \overrightarrow{A}_{r1,g} + f.(\overrightarrow{A}_{r2,g} - \overrightarrow{A}_{r3,g}) + f.(\overrightarrow{A}_{r4,g} - \overrightarrow{A}_{r5,g}) \tag{11}$$

Zhang et al. [52] proposed an adaptive differential evolution with an optional external archive and considered both the best global solution with the current solution that can be formulated as follows.

$$DE/current-to-pbest/1/bin : \overrightarrow{V}_{i,g} = \overrightarrow{A}_{i,g} + f.(\overrightarrow{A}^{p}_{best,g} - \overrightarrow{A}_{i,g}) + f.(\overrightarrow{A}_{r1,g} - \overrightarrow{A}_{r2,g}) \tag{12}$$

The second evolutionary operator in DE is crossover. The most frequent type of crossover method is binomial, and it can be formulated as follows:

$$\overrightarrow{U}^{g}_{i,j} = \begin{cases} \overrightarrow{V}^{g}_{i,j}, & \text{if (rand} \le CR) \text{ or } (j = sn), \\ \overrightarrow{A}^{g}_{i,j}, & \text{otherwise.} \end{cases} \quad j = 1, 2, \ldots D \tag{13}$$

where U^g is the trial vector, and CR is the probability crossover rate and can be between zero and one. sn is the number of candidates selected in the process of crossover. Finally, a greedy combination of the new solution made and its parent is produced to replace the offspring.

$$\overrightarrow{A}^{g+1}_{i} = \begin{cases} \overrightarrow{U}^{g}_{i}, & \text{if } \left(fit\left(\overrightarrow{U}^{g}_{i}\right) \le fit\left(\overrightarrow{A}^{g}_{i}\right)\right), \\ \overrightarrow{A}^{g}_{i}, & \text{otherwise.} \end{cases} \tag{14}$$

2.5.2. Covariance matrix adaptation evolution strategy (CMA-ES)

Hanson et al. [53] proposed a new insight into the evolutionary algorithms and introduced the derandomised Evolution Strategy with Covariance Matrix Adaptation (CMA-ES). CMA-ES is a well-known, fast [54], and effective population-based, gradient-free optimisation approach. It does not require any initial control parameters to be set, and it is able to adjust its control parameters in the form of self-adaptation. CMA-ES performance has been considered in various black-box and real engineering optimisation problems with diverse characteristics , including continuous, discrete, ill-conditioned, constrained, multi-modal and non-separable problems [55].

In CMA-ES, a multivariate normal distribution function ($A_i \sim \mathcal{N}(\mu^g, \xi^g)$) is used to generate n-dimensional solutions $A_i \in \mathbb{R}^n$ at each

iteration as follows:

$$A_i^{g+1} \sim \mathcal{N}(\mu^{(g)}, (\sigma^{(g)})^2 C^{(g)}) \forall \quad i \in \{1, \dots, n\} \tag{15}$$

where μ^g and C^g are the average and covariance matrix of the solutions in the previous population, respectively. σ^g and n are the search step size and the population size. After generating the samples based on Eq. (15) and evaluating them using the objective function, they will be sorted and used to estimate ξ^{g+1}, μ^{g+1}, and σ^{g+1}.

2.5.3. Generalised normal distribution optimisation (GNDO)

Zhang et al. [56] proposed a fast and robust meta-heuristic algorithm entitled generalised normal distribution optimisation (GNDO), motivated by the hypothesis of normal distribution (suitable for a probability bell curve) and performing well in global optimisation problems. The main components of GNDO are local exploitation and global exploration. Three solutions are randomly selected in the global exploration strategy to make a triangle search pattern. The normal distribution pattern tries to be generalised in local exploration using the existing local optimum and the average of the solutions in the current population. Explicit characterisations of local and global learning techniques are as follows:

Local exploitation. In the local search process of GNDO, the main focus is on exploring the surroundings of the candidates' location to find better solutions in terms of efficiency. Eq. (16) illustrates how the developed normal distribution can be helpful in searching the objective space based on the distribution of the solution in the current population.

$$\overrightarrow{V}_k^r = \mu_k + (\sigma_k \times P_f), k = 1, 2, \dots, N_{pop} \tag{16}$$

where $\overrightarrow{V}_k^r$ is the vector of GNDO's mutation operator. The solution number and iteration are shown by k and r, respectively. μ_i, δ_i and P_f are the generalised means, the standard variance, and the penalty factor, respectively. To have a better understanding of these three parameters, the formulations are as

$$\mu_k = \frac{(A_{Best}^r + A_k^r + M_l)}{3} \tag{17}$$

$$\delta_k = \sqrt{\frac{((A_{Best}^r - \mu)^2 + (A_k^r - \mu)^2 + (M_l - \mu)^2)}{3}} \tag{18}$$

$$\eta = \begin{cases} \sqrt{-\log(\theta_1)}.sin(90^\circ - 2\pi\theta_2), if(x1 \le x2) \\ \sqrt{-\log(\theta_1)}.sin(90^\circ - (2\pi\theta_2 + \pi)), otherwise \end{cases} \tag{19}$$

where $x1, x2, \theta_1$ and θ_2 are random variables [0-1]. The best-found solution is shown by A_{Best}^r, and M_l is the current population's position average formulated as follows.

$$M_l = \frac{1}{N_{pop}} \sum_{k=1}^{N_{pop}} A_k^r \tag{20}$$

Global exploration. In the standard GNDO, the global exploration approach can be described as follows:

$$\overrightarrow{V}_k^r = (\overrightarrow{A}_k^r) + [\alpha.(|\theta_3| \times \overrightarrow{V}_{r1}^r)] + [(1-\alpha).(|\theta_4| \times \overrightarrow{V}_{r2}^r)] \tag{21}$$

where θ_3 and θ_4 are random variables associated with normal distribution, α is defined based on a range of 0 and 1 randomly. Finally, $\overrightarrow{V}_{r1}^r$ and $\overrightarrow{V}_{r2}^r$ are two vectors generated by the mutation operator by:

$$\overrightarrow{V}_{r1} = \begin{cases} \overrightarrow{A}_k^r - \overrightarrow{A}_{s1}^r, if(f(\overrightarrow{A}_k^r) < f(\overrightarrow{A}_{s1}^r)) \\ \overrightarrow{A}_{s1}^r - \overrightarrow{A}_k^r, otherwise \end{cases} \tag{22}$$

$$\overrightarrow{V}_{r2} = \begin{cases} \overrightarrow{A}_{s2}^r - \overrightarrow{A}_{s3}^r, if(f(\overrightarrow{A}_{s2}^r) < f(\overrightarrow{A}_{s3}^r)) \\ \overrightarrow{A}_{s3}^r - \overrightarrow{A}_{s2}^r, otherwise \end{cases} \tag{23}$$

where s_1, s_2 and s_3 are three solutions randomly selected from the population, which should satisfy the restriction of $s1 \neq s2 \neq s3 \neq k$. The vector of $\overrightarrow{V}_{r1}$ involves the local learning term, and there is a direct interaction between $s1$ and kth. To consider a global information interaction mechanism, the vector of $\overrightarrow{V}_{r2}$ is introduced, and kth solution contributes with the other two solutions s_2 and s_3.

α plays the role of a stabiliser parameter to provide an appropriate balance between local and global search. θ_3 and θ_4 are initialised based on a random number from the normal distribution, resulting in a more developed global search ability in GNDO.

2.5.4. Arithmetic optimisation algorithm (AOA)

One modern and successful meta-heuristic method is the Arithmetic Optimisation Algorithm (AOA) [57]. AOA is based on a population of feasible candidates and four simple arithmetic operations, division, multiplication, subtraction, and addition. The initial population (A) is randomly generated as follows.

$$A = \begin{bmatrix} a_1^1 & a_1^2 & \dots & a_1^d \\ a_2^1 & a_2^2 & \dots & a_2^d \\ \vdots & \vdots & & \vdots \\ a_{N_{pop}}^1 & a_{N_{pop}}^2 & \dots & a_{N_{pop}}^d \end{bmatrix} \tag{24}$$

where N_{pop} is the size of population, and d is related to problem dimension. The best-found solution in the current population is compared with the previous global best solutions, and the best one is kept as nearly optimum.

In Eq. (25), a Mathematical Optimiser Accelerated (MOA) technique is computed to develop a better balance between AOA's exploration or exploitation search capabilities.

$$MO(I_{it}) = Min_{val} + I_{it} \times (\frac{Max_{val} - Min_{val}}{Max_{it}}) \tag{25}$$

where I_{it} and Max_{it} are the current replication number and the maximum number of evaluations. The maximum and minimum fitness are shown by the Min_{val} and Max_{val}, respectively.

Eq. (26) shows the formulation of the exploration phase of AOA. Various areas of the objective search space are effectively explored by this operator. Division and multiplication math operators are implemented to develop the exploration ability of AOA.

$$a_{k,j}(I_{it+1}) = \begin{cases} (a_j^{best}) \div (MO + \gamma) \times ((upper_j - lower_j) \times \mu + lower_j) & r_2 < 0.5 \\ (a_j^{best}) \times MO \times ((upper_j - lower_j) \times \mu + lower_j) & otherwise \end{cases} \tag{26}$$

where $a_k(I_{it+1})$ and $a_{k,j}(I_{it})$ are the kth candidate in the next population, and jth decision variable of the kth solution at the current population. A small integer value is assigned to γ, and $upper_j$ and $lower_j$ are the lower and upper bounds of the search space, respectively. Furthermore, an adjusted parameter was introduced ($\mu = 0.5$) to control the search speed properly.

$$MO(I_{it}) = 1 - (I_{it})^{\frac{1}{\xi}} / (Max_{it})^{\frac{1}{\xi}} \tag{27}$$

where mathematical optimiser probability (MO) recreates as a coefficient function, ξ presents a sensitive factor, defines the exploitation strategy's accurateness over the search process, and initialises at five.

In the last step of AOA, subtraction and addition operators are used to improve the exploitation process. Eq. (28) shows this formulation.

$$a_{i,j}(I_{it+1}) = \begin{cases} (a_j^{best}) - (MO) \times ((upper_j - lower_j) \times \mu + lower_j) & r_3 < 0.5 \\ (a_j^{best}) + (MO) \times ((upper_j - lower_j) \times \mu + lower_j) & otherwise \end{cases} \quad (28)$$

Algorithm 1 The Adaptive Covariance Differential Evolution(α)

procedure The Adaptive Covariance Differential Evolution(α)
Initialization
Pop_{Size}(NP), other control parameters
$P_\sigma = P_c$=0, C=1, $B = I_n$, D=ones(D,1), $\sigma = 0.5$
Initialize the first population by the following formula: $Pop_1 = unifrnd(VarMin, VarMax, [Pop_{size}1])$;
while $g \le Max_{Iter}$ **do**
Updating Evolutionary Parameters
$m, P_\sigma, P_c, C, \sigma$ are updated.
Computing B, D through Eigen decomposition of C.
$C^{1/2}, P, F$(scale factor) are obtained.
$F_g = 0.2 + 0.5 * rand(0,1)$
$C_o = Min_c + iter \times ((Max_c - Min_c)/iter)$
while $i \le NP$ **do {Mutation}**
Randomly select five solutions from current population $\vec{X}_{r_1,g}, ..., \vec{X}_{r_5,g}$ then choose the most successful mutated strategies and generate the mutated vector (assumed (ST_1)):

$$\vec{V}_{i,g} = \begin{cases} \vec{A}_{r_1 i} + F.(\vec{A}_{r_2 i} - \vec{A}_{r_3 i}) & \alpha_1 \ge C_o \ \& \ \alpha_2 \le 0.5 \\ \vec{A}_i + F.(\vec{A}_{r_1 i} - \vec{A}_{r_2 i}) & \alpha_1 \ge C_o \ \& \ \alpha_2 > 0.5 \\ \vec{A}_{best} + F.(\vec{A}_{r_1 i} - \vec{A}_{r_2 i}) & \alpha_1 < C_o \ \& \ \alpha_2 \le 0.5 \\ \vec{A}_{best} + F.(\vec{A}_{r_1 i} - \vec{A}_{r_2 i}) + F.(\vec{A}_{r_3 i} - \vec{A}_{r_4 i}) & \alpha_1 < C_o \ \& \ \alpha_2 > 0.5 \end{cases} +$$

$\sigma.B.D.randn(D)^T$
end while
while $j \le N_{var}$ **do {Crossover}**
Generate $j_{rand} = ceil(rand(1, D))$,and C_r=Eq.(32)
if $rand_{i,j} \le C_r$ or $j = j_{rand}$ **then** $v_{j,g} \to U_{j,g}$
else $X_{j,g} \to U_{j,g}$
end if
end while
{Selection}
if $f(\vec{U}_{i,g}) \le f(\vec{X}_{i,g})$ **then** $U_{i,g} \to X_{i,g+1}$
else $X_{i,g} \to X_{i,g+1}$
end if
g++
end while
end procedure

In terms of the quality of the best-found candidates and the convergence speed, AOA's performance is considerable. Moreover, AOA offers the appropriate capacity to avert trapping of the local optima.

2.5.5. Multi-strategy differential covariance matrix evolutionary algorithm (ADCMA)

Recently, multi-strategy evolutionary algorithms have been developed and used in a wide range of real engineering optimisation problems [58] due to their high abilities in exploration and exploitation. The standard CMA-ES method applies a covariance matrix and is updated adaptively to specify the objective function search space. Although CMA-ES performance is considerable, it can encounter premature convergence and local optima in the most complex, hybrid and noisy problems [59,60]. To improve this significant demerit of CMA-ES, a multi-strategy differential perturbation technique has been introduced. The new mutated vector is created in this algorithm by the following equation.

$$\vec{V}_{i,g} = \begin{cases} \vec{A}_{r_1 i} + F.(\vec{A}_{r_2 i} - \vec{A}_{r_3 i})(1) & \alpha_1 \ge C_o \ \& \ \alpha_2 \le 0.5 \\ \vec{A}_i + F.(\vec{A}_{r_1 i} - \vec{A}_{r_2 i})(2) & \alpha_1 \ge C_o \ \& \ \alpha_2 > 0.5 \\ \vec{A}_{best} + F.(\vec{A}_{r_1 i} - \vec{A}_{r_2 i})(3) & \alpha_1 < C_o \ \& \ \alpha_2 \le 0.5 \\ \vec{A}_{best} + F.(\vec{A}_{r_1 i} - \vec{A}_{r_2 i}) & \\ \quad + F.(\vec{A}_{r_3 i} - \vec{A}_{r_4 i})(4) & \alpha_1 < C_o \ \& \ \alpha_2 > 0.5 \end{cases} + \sigma.B.D.randn(D)^T \quad (29)$$

where r_1, r_2, r_3 and r_4 are exclusive integers randomly selected from the range of $[1 - N_{pop}]$ with a constraint $r_1 \ne r_2 \ne r_3 \ne r_4$. α_1 and α_2 are random numbers chosen between zero and one. The mutation factor F is a positive random number generated by Eq. (30). C_o is a control parameter that linearly increases from Min_c to Max_c (See Eq. (31)) to adjust the balance between exploration and exploitation. D is the dimension of the problem, and $randn(D)$ is a vector of random numbers that comes from a $\mathcal{N}(0,1)$. σ is initialised by 0.5 and updated each generation using the covariance matrix. B is an orthonormal basis of eigenvectors to make an eigendecomposition of the covariance matrix.

$$F_i = 0.2 + 0.5 * rand(0,1) \quad (30)$$

$$C_o = Min_c + iter \times ((Max_c - Min_c)/iter) \quad (31)$$

The main reason for selecting four differential covariance matrix strategies is that the initial two strategies strongly improve the global searchability of ADCMA and develop a robust exploration in the optimisation procedure. On the other hand, the third and fourth mutation strategies developed based on the best solution so far reinforce the exploitation behaviour at the end of the ADCMA optimisation process. The introduced control parameter C_o provides a chance to select each mutation strategy with a different probability rate during each iteration. This means that we have the benefits of exploration and exploitation searchability of all four strategies together.

A binomial crossover is applied to combine the mutated vectors with their parents in each generation. The crossover probability rate C_r is made by a normal distribution with a small standard deviation.

$$Cr = Gaussian(Cr_m, 0.1) \quad (32)$$

This study used a competitive selection process (similar to DE) based on comparing the newly generated offspring with its parent. If the fitness of the progeny dominates the parent's fitness, then it will be selected to play a role in the next generation; otherwise, the parent will be kept. The proposed adaptive covariance DE algorithm can be seen in Algorithm 1.

2.6. The proposed solar radiation prediction framework

The following section presents the framework of the proposed hybrid model and its particular implementation for forecasting solar radiation based on historical meteorological wind speed, direction, temperature, and Shortwave Radiation (SRAD1) data from an installed offshore buoy station in Lake Michigan, Chicago, United States [61].

1. Pre-processing: In order to improve the performance of the proposed models various pre-processing studies were employed, namely cleaning and replacing the missing value, detecting and de-noising outliers (autoencoder), and normalising various features in the range of zero and one.
2. Feature selection: Nine LSTM models with different input configurations were developed and compared, which can be seen in Fig. 10. The best-performing model was system 1 with four inputs: wind speed, direction, temperature and SRAD.
3. Decomposition: An adaptive multivariate empirical mode decomposition (AMEMD) algorithm was proposed to decompose solar radiation time-series data with a high level of nonlinearity and non-stationarity into various IMFs with minimum entropy using an evolutionary Nelder–Mead simplex search algorithm. Fig. 2 shows the optimisation process of permutation entropy of the AMEMD.
4. Forecasting: In order to recognise the dependencies of meteorological and solar radiation time-series data, four hybrid deep learning models were developed consisting of an LSTM, bi-directional LSTM, stacked LSTM and a gated recurrent unit

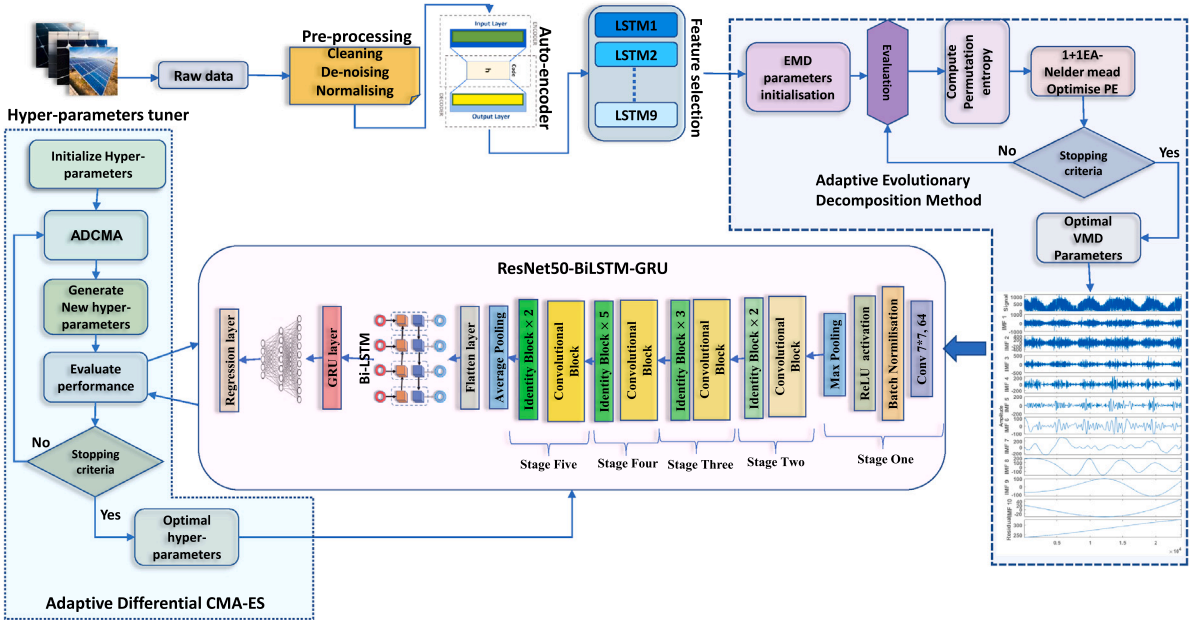

Fig. 6. A schematic representation of the proposed hybrid solar radiation forecasting.

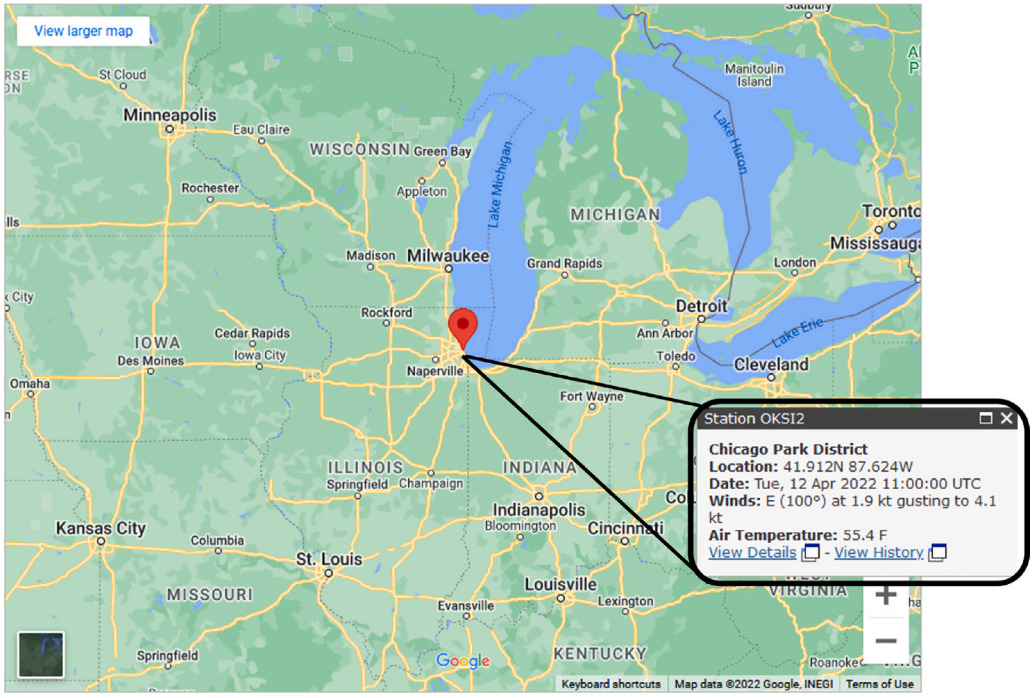

Fig. 7. The geographical details of the station OKSI2 - Oak St., Chicago, IL (National Data Buoy Center) and recorded solar radiation and Standard meteorological data used in this study [62].

model. The general architecture of these models consists of a sequence input layer, two or three layers of LSTM/BiLSTM/GRU, and dropout layers with a coefficient of 0.1. A fully connected layer with a regression layer was embedded in the pursuit to yield the last predicted solar radiation rate.

5. Feature extraction: To extract more effective features from the meteorological and solar radiation time-series data to boost the effectiveness of the hybrid model, ten convolutional deep models were developed and combined with the recurrent deep models. In the initial five CNN layers, Dropout, Max-pooling layers, and after a flattened layer, there was one fully connected layer to finalise the outputs. Furthermore, two popular and modern deep learning models, Xception and ResNet50, were combined with the GRU and LSTM models to build a robust and accurate forecasting model.
6. Hyper-parameter tuning: to find the optimal hyper-parameters of the hybrid forecasting model (ResNet50-GRU-2LSTM), we proposed a novel optimiser: Multi-strategy differential covariance matrix evolutionary algorithm (ADCMA). The performance of ADCMA compared with five state-of-the-art and popular optimisation algorithms, including DE, AOA, GNDO, and CMA-ES for all feasible configurations of 10 hyper-parameters, batch size,

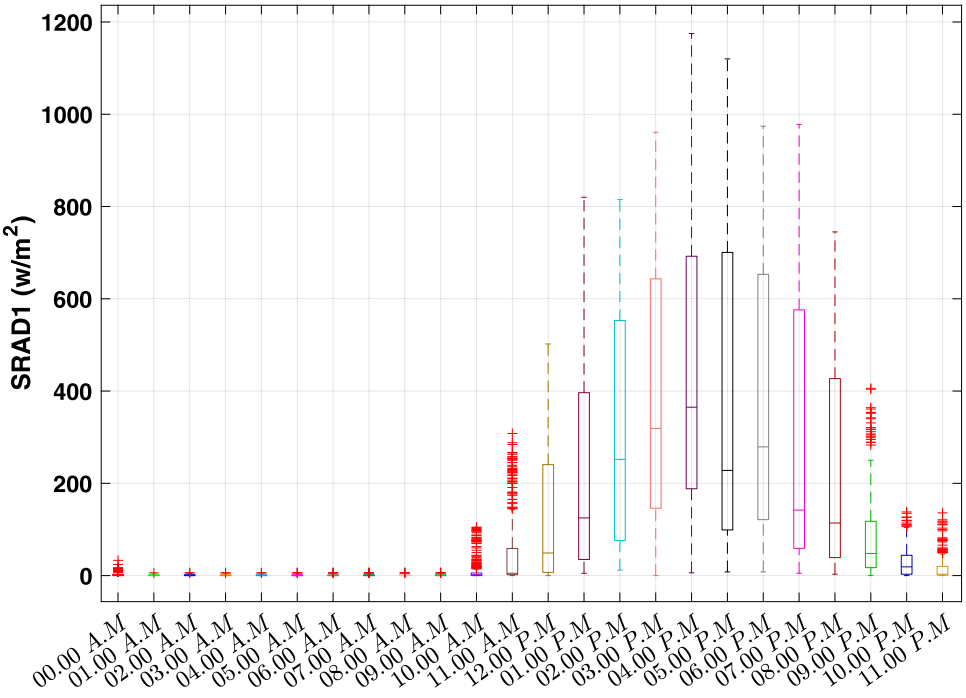

Fig. 8. The boxplot of SRAD1 distributions measured at the OKSI2 station per hour during 2016.

Table 3
Explanatory statistics of average shortwave radiation in watts per m^2 (SRAD1 is from an LI-COR LI-200 pyranometer sensor) for OKSI2 station. The sampling frequency stands two times per second (2 Hz).

	2014	2015	2016	2017	2018	2019	2020	2021	2022
Min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Max	898.00	1064.00	1175.00	1120.00	1104.00	1019.00	1122.00	1129.00	1033.00
Mean	254.20	272.96	271.31	290.78	291.18	427.96	261.61	268.10	263.10
Median	196.38	187.89	177.00	210.00	205.69	385.75	176.63	186.63	167.00
STD	223.38	252.60	256.32	258.61	263.66	256.06	248.71	248.63	254.13
Interquartile	333.00	372.50	378.94	406.88	405.25	467.00	366.50	367.50	347.88
Skewness	0.81	0.89	0.91	0.77	0.77	0.26	0.94	0.89	1.03

learning rate, convolutional filter number and filter size, dropout layer coefficient, weight initialiser, solver method, LSTM and GRU hidden size.

To show a more detailed landscape of the proposed hybrid forecasting model and its various subsections, Fig. 6 is indicated.

3. Case study

In this study, we considered the collected real dataset consisting of a combination of solar radiation and standard meteorological data collected from Station OKSI2 - Oak St., Chicago, IL (National Data Buoy Center), from January 2014 to June 2022. The time resolution of data collection was one hour. Fig. 7 shows the geographical location of the station with the online wind speed and air temperature. As the solar radiation data includes many zero values related to nights (can be indicated in Fig. 8), we should filter the helpful time steps from the raw data. Therefore, we select the recorded data between 12 P.M. and 20 P.M. for training and validating the predictive model. To attain an in-depth investigation of the dataset, statistical analyses for solar radiation between 2014 and 2022 were applied, which can be seen in Table 3. The highest solar radiation months are from April to September, with an average rate of more than 400 (w/m^2), which is shown in Fig. 9(a). Furthermore, the distribution of wind direction and wind speed are visualised in Fig. 9(b) and (c).

4. Experiments and analysis

4.1. Backbone architecture of recurrent deep learning model

In the first step of this study, nine LSTM models with various features (with one, two, three, and four inputs of SRAD, wind speed, wind direction and air temperature) were proposed and compared to find the best model for the prediction accuracy of solar radiation. A schematic diagram of nine LSTM forecasting models with a representation of the inputs can be seen in Fig. 10. Moreover, We can see the application of LSTM in Fig. 10 as a feature selection technique involves training an LSTM model, interpreting the learned weights or feature importances, selecting the most critical features, training a new model (a combination of LSTM and GRU with a grid search for hyper-parameters tuning) using the selected features, and evaluating the performance of the new model. This approach can be helpful in reducing the dimensionality of the data and improving the model's performance. Meanwhile, the statistical analysis of the solar prediction accuracy for nine LSTM models can be depicted in Fig. 11 and Table 4, where each model runs ten times independently. From this box plot, the best-performed LSTM model is system 1, with four inputs of time-series data and minimum average learning error. These modelling results (from Fig. 11) show that the proposed feature selection method selects all features, which means that none of the components was considered redundant or irrelevant to the target variable. In other words, all the

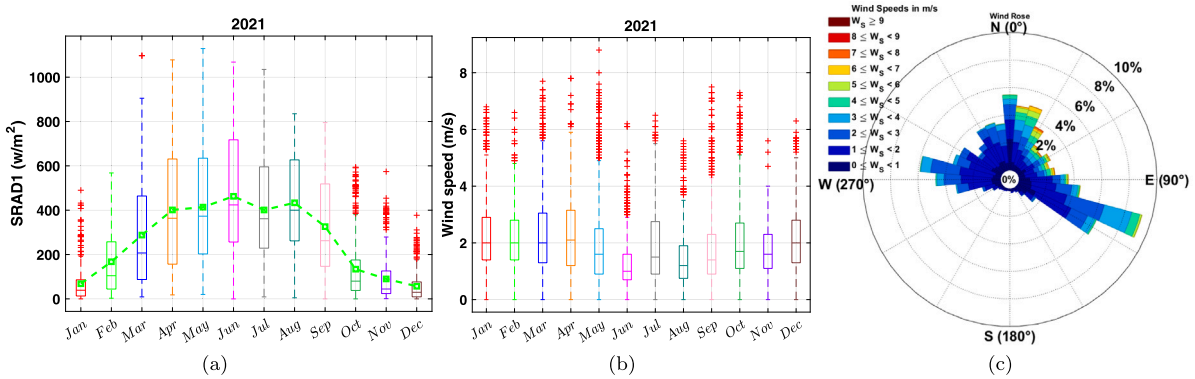

Fig. 9. (a) Monthly distribution of SRAD1 values between 12 P.M and 20 P.M measured at the OKSI2 station in 2021 (b) wind speed distribution for 12 months in 2021, (c) Wind direction, speed and frequency at OKSI2 station in 2021.

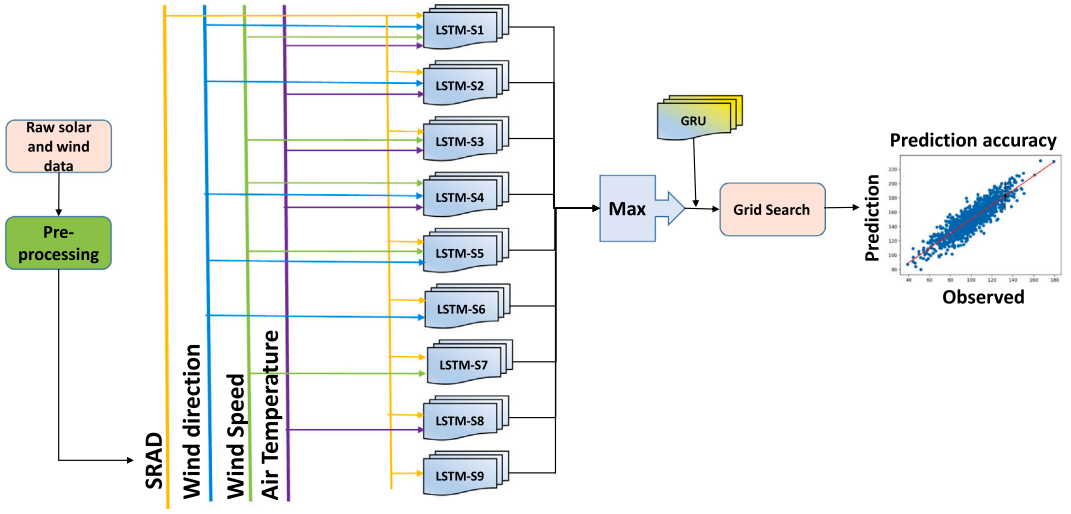

Fig. 10. A schematic diagram of nine LSTM forecasting models with a representation of the inputs.

Table 4
The average SRAD prediction accuracy of eight LSTM models with various inputs. The batch size and learning rate are 254 and 10^{-5}, respectively.

	LSTM-S1	LSTM-S2	LSTM-S3	LSTM-S4	LSTM-S5	LSTM-S6	LSTM-S7	LSTM-S8	LSTM-S9
MSE	**1.5205E−02**	1.5487E−02	1.5208E−02	1.5216E−02	3.6139E−02	1.5535E−02	1.5498E−02	1.5222E−02	1.5532E−02
RMSE	**1.2331E−01**	1.2445E−01	1.2332E−01	1.2335E−01	1.9010E−01	1.2464E−01	1.2449E−01	1.2338E−01	1.2463E−01
R-value	**8.1845E−01**	8.1558E−01	8.1843E−01	8.1839E−01	4.7532E−01	8.1510E−01	8.1561E−01	8.1830E−01	8.1513E−01
MAE	**8.5566E−02**	8.7069E−02	8.5652E−02	8.5869E−02	1.5780E−01	8.7299E−02	8.7193E−02	8.5905E−02	8.7198E−02

features were deemed essential and helpful in predicting the target variable. Totally, this can occur when the dataset is small or when the features are highly correlated with the target variable. In such cases, the model may function adequately when all the features are employed, as releasing any of them may cause a loss of information and result in inferior performance.

In the following, we tested and compared three different model training optimisers; 'sgdm', 'adam', and 'rmsprop' for an LSTM model (BS=256 and LR=10^{-5}). Fig. 12(a) indicates that both rmsprop and adam performance are better than sgdm and are the same approximately.

In the second phase, we proposed four hybrid solar radiation models, LSTM-GRU, BiLSTM-GRU, GRU-LSTM and 2GRU-BiLSTM. The performance of these four proposed models was compared with five popular machine learning methods (feed-forward neural network (FFNN), adaptive neural fuzzy inference system (ANFIS), LSTM, BiLSTM, and GRU) and three stacked recurrent models, S-LSTM, S-BiLSTM and S-GRU.

The experimental results show that a combination of GRU and LSTM models can perform better than other hybrid models because of using the characteristic of both models together. Among the three stacked models, S-GRU's performance is considerable compared with that of S-LSTM and S-BiLSTM. The whole comparative ML framework performance can be seen in Fig. 13. From this figure, the best-performing model on average is LSTM-GRU. Furthermore, the performance of the S-GRU and GRU-LSTM models is also competitive.

Table 5 reports the statistical validation results of the solar radiation forecasting analysis for 12 studied learning models in terms of MSE, RMSE, R-value and MAE. It can be seen that the hybrid GRU-LSTM could propose the minimum validation learning error (MSE), which performed considerably better than standard LSTM, GRU, stacked LSTM, and stacked GRU by 18%, 5%, 10%, and 1.3%. The main technical reasons for the high performance of the hybrid GRU-LSTM model compared with other hybrid machine learning models in forecasting short-term solar radiation are:

(i) The capability of capturing long-term dependencies in the input data may be the most important attribute of the hybrid GRU-LSTM

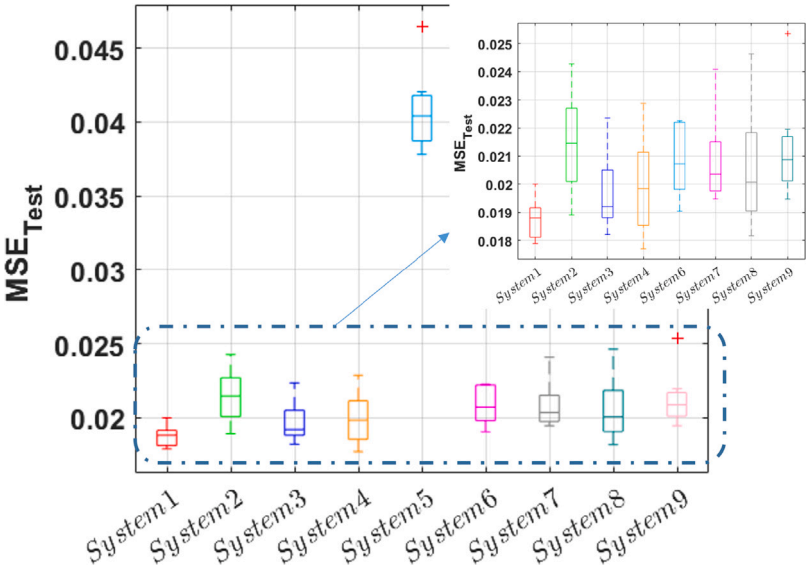

Fig. 11. The MSE average OF ten runs of nine forecasting models (batch size=256, learning rate=10^{-5}, Neuron number=200, maxEpochs=150, Optimizer Name='adam').

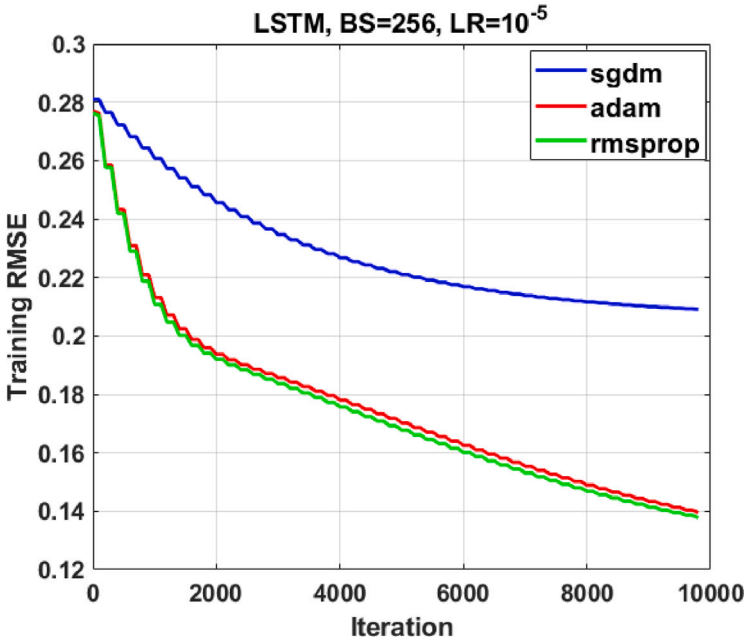

Fig. 12. (a) a comparison of three optimisation training methods: sdam, adam, and rmsprop.

model, which is crucial for accurately forecasting solar radiation. The LSTM component of the model is particularly effective at capturing long-term dependencies, while the GRU component is better suited for capturing short-term dependencies [63].

(ii) This hybrid model can handle variable-length input sequences, which is essential for forecasting solar radiation as the number of input features and the length of the input sequence can vary depending on the time of day and weather conditions.

(iii) The GRU-LSTM model is robust to noisy data, which can be vital for forecasting solar radiation as weather conditions can be unpredictable and noisy. The model is able to filter out noise in the input data and focus on the most critical features for forecasting solar radiation. However, the most considerable hybrid model is GRU-BiLSTM in terms of the correlation coefficient between the target (SRAD) and predicted data. In addition, the correctness of the solar prediction of the stacked models (S-LSTM, S-BiLSTM and S-GRU) is better than that of the vanilla versions, except for Bi-LSTM. Last but not least, combining the GRU model as a pre or post-layer could improve both average learning error and prediction accuracy.

As the impact of the learning rate on the forecasting model can be meaningful [64], in this phase, we developed a simple grid search to evaluate the importance of the learning rate and batch size hyper-parameters of the GRU-2BiLSTM model. The results of this analysis can be seen in Fig. 14. As can be marked, the best range of learning rate is around 10^{-4}; however, the batch size plays a significant role in providing a considerable performance (greater than 256). Indeed, the learning rate and batch size are important hyperparameters in training LSTM and GRU models and are able to provide a multi-model search landscape that can be seen in Fig. 14. A reasonable learning rate can enable the model to converge speedily and achieve more reasonable

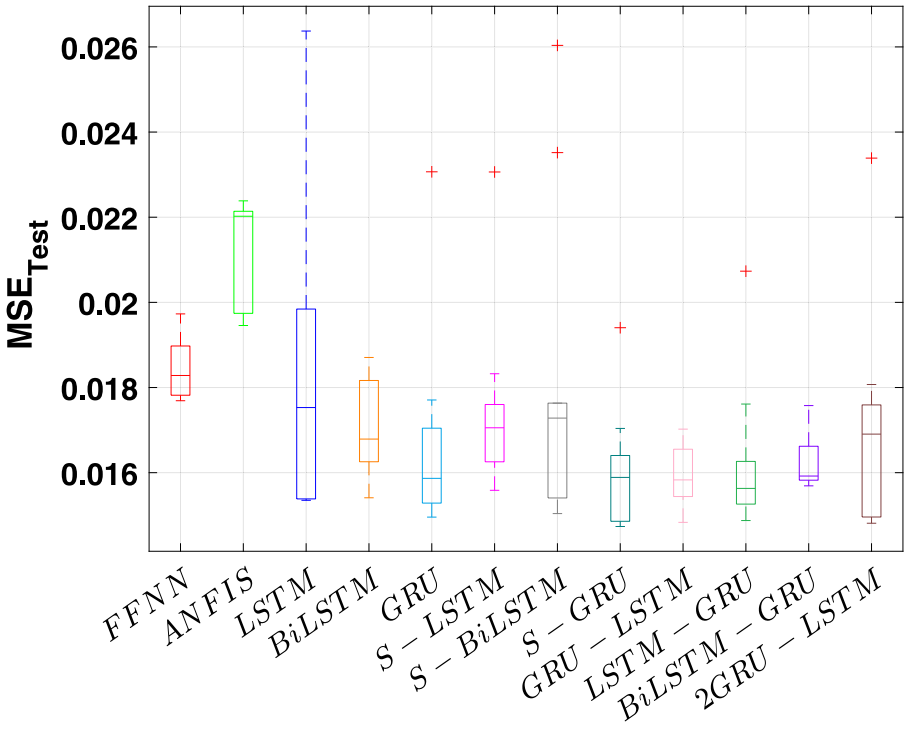

Fig. 13. The average MSE of the validation of 12 machine learning models for predicting solar radiation.

Table 5
The average SRAD prediction accuracy of ten independent runs for five popular machine learning models, three stacked recurrent models and four hybrid deep learning models. The batch size and learning rate are 254 and 10^{-5}, respectively.

	FFNN					ANFIS					LSTM				
	Mean	Min	Max	Median	STD	Mean	Min	Max	Median	STD	Mean	Min	Max	Median	STD
MSE	0.0185	0.0177	0.0197	0.0183	0.0007	0.0213	0.0195	0.0224	0.0220	0.0012	0.0187	0.0154	0.0264	0.0175	0.0042
RMSE	0.1358	0.1330	0.1405	0.1351	0.0026	0.1438	0.1364	0.1481	0.1468	0.0049	0.1336	0.1215	0.1564	0.1315	0.0131
R-value	0.7750	0.7010	0.7988	0.7921	0.0385	0.7009	0.6151	0.7377	0.7277	0.0476	0.7352	0.6619	0.7834	0.7484	0.0461
MAE	0.1024	0.0970	0.1086	0.1016	0.0041	0.1021	0.0962	0.1064	0.1044	0.0040	0.1008	0.0911	0.1215	0.0978	0.0115
	BiLSTM					GRU					S-LSTM				
	Mean	Min	Max	Median	STD	Mean	Min	Max	Median	STD	Mean	Min	Max	Median	STD
MSE	0.0171	0.0154	0.0187	0.0165	0.0012	0.0167	0.0150	0.0231	0.0159	0.0024	0.0175	0.0156	0.0231	0.0171	0.0021
RMSE	0.1294	0.1215	0.1358	0.1275	0.0050	0.1272	0.1195	0.1479	0.1244	0.0083	0.1305	0.1220	0.1487	0.1295	0.0075
R-value	0.7517	0.7045	0.7828	0.7589	0.0282	0.7236	0.6713	0.7811	0.7061	0.0412	0.7279	0.6670	0.7798	0.7174	0.0432
MAE	0.0969	0.0919	0.1021	0.0955	0.0036	0.0926	0.0863	0.1105	0.0914	0.0068	0.1001	0.0929	0.1154	0.0989	0.0060
	S-BiLSTM					S-GRU					GRU-LSTM				
	Mean	Min	Max	Median	STD	Mean	Min	Max	Median	STD	Mean	Min	Max	Median	STD
MSE	0.0182	0.0150	0.0260	0.0173	0.0037	0.0161	0.0147	0.0194	0.0159	0.0014	**0.0159**	0.0148	0.0170	0.0158	0.0008
RMSE	0.1321	0.1205	0.1550	0.1306	0.0115	0.1269	0.1192	0.1375	0.1241	0.0057	**0.1249**	0.1198	0.1308	0.1244	0.0039
R-value	0.7379	0.6671	0.7897	0.7521	0.0521	0.7409	0.6871	0.7872	0.7490	0.0359	0.7496	0.6902	0.7848	0.7500	0.0334
MAE	0.0972	0.0881	0.1172	0.0950	0.0097	0.0884	0.0845	0.0986	0.0875	0.0043	0.0899	0.0859	0.0935	0.0898	0.0026
	LSTM-GRU					GRU-BiLSTM					2GRU-LSTM				
	Mean	Min	Max	Median	STD	Mean	Min	Max	Median	STD	Mean	Min	Max	Median	STD
MSE	0.0162	0.0149	0.0207	0.0156	0.0018	0.0163	0.0157	0.0176	0.0159	0.0008	0.0170	0.0148	0.0234	0.0169	0.0026
RMSE	0.1254	0.1192	0.1414	0.1233	0.0067	0.1266	0.1237	0.1316	0.1250	0.0033	0.1285	0.1196	0.1484	0.1293	0.0089
R-value	0.7250	0.6776	0.7810	0.7297	0.0386	**0.7614**	0.7286	0.7894	0.7586	0.0250	0.7489	0.6887	0.7861	0.7628	0.0398
MAE	0.0904	0.0862	0.1023	0.0889	0.0047	0.0907	0.0890	0.0938	0.0897	0.0020	0.0922	0.0856	0.1080	0.0922	0.0067

accuracy, while a suitable batch size can balance training speed and stability. Experimenting with different values of these hyperparameters is essential to find the optimal deals for a given dataset and model architecture.

4.2. Hybrid adaptive Xception and Resnet50 with GRU and BiLSTM

In this section, we tested and compared the efficiency of various hybrid convolutional recurrent deep learning models in forecasting solar radiation one hour ahead. The prediction accuracy and learning error were measured by mean square error (MSE), root MSE (RMSE), correlation coefficient (R-value), mean absolute error (MAE), and symmetric mean absolute percentage error (SMAPE). The SMAPE is calculated as follows.

$$SMAPE = \frac{1}{n}\sum_{t=1}^{n}\frac{|T_t' - T_t|}{\frac{T_t' + T_t}{2}} \tag{33}$$

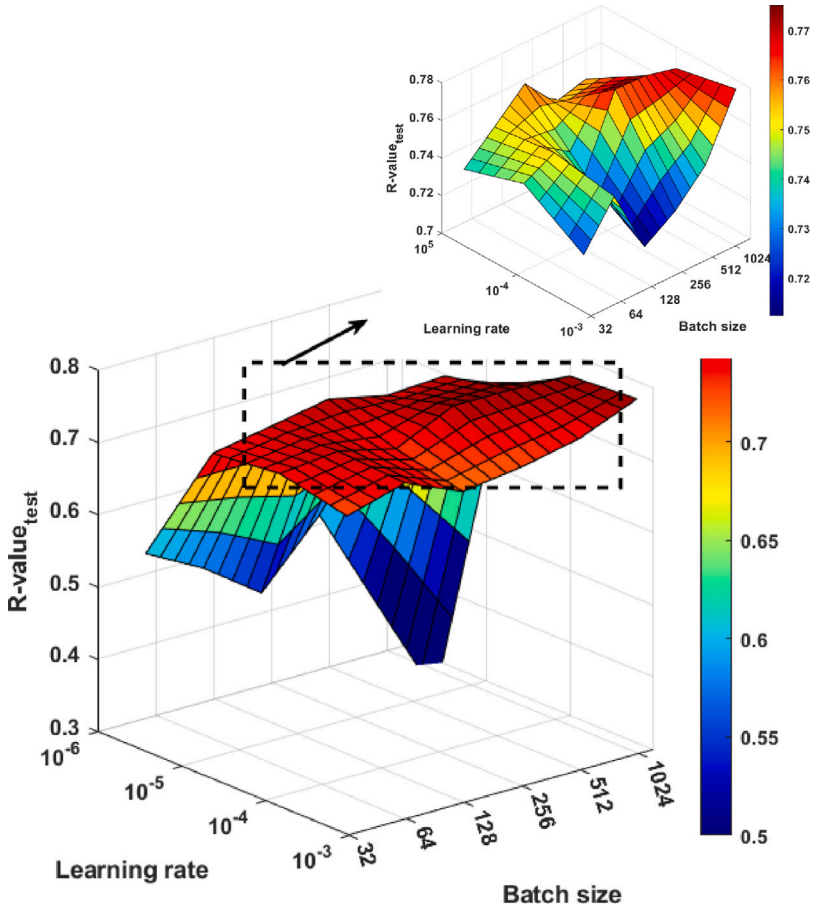

Fig. 14. Hyper-parameter landscape analysis of the GRU-2BiLSTM model's performance with four inputs.

where n is the total number of samples and T' and T stand for the predicted and target sample, respectively.

In order to evaluate the impact of convolutional neural networks combined with three RNN models, LSTM, BiLSTM, and GRU, we proposed a primary deep learning model with five convolutional layers and three recurrent layers (5CNN-3LSTM, 5CNN-3BiLSTM, and 5CNN-3GRU). The statistical results showed that the 5CNN-3LSTM performed better than 5CNN-3GRU and 5CNN-3BiLSTM by 3.2% and 1.3% in terms of MSE that can be seen in Table 6. Moreover, we analysed the performance of two embedded models in the hybridisation of convolutional, LSTM and GRU layers (5CNN-GRU-2LSTM and 5CNN-2GRU-LSTM). From Table 6, we can see that the developed model with two LSTM layers could propose a predicted solar radiation value with a lower learning error (at 3.3%). To explore the optimal number of convolutional layers in 5CNN-GRU-2LSTM, we removed one convolutional layer at each step and tested the performance of the hybrid model iteratively until only one convolutional layer remained. The modelling achievements demonstrated that reducing the convolutional layers decreased the hybrid model's forecasting ability and robustness (see Table 6, second and third row, and Fig. 15). Finally, two hybrid advanced convolutional models were tested, Xception [45] (170 layers with 22.8 million learnable parameters) and ResNet50 [47] (177 layers with over 23 million trainable parameters). It can be seen that a combination of ResNet50 and LSTM was more accurate than Xception+LSTM at 1.4%, and adding a GRU layer (ResNet50-GRU-2LSTM) enhanced this improvement by 5.8% based on the validation MSE.

Figs. 15 and 16 display the statistical analysis comparisons of the ten independent training and testing runs for 11 hybrid models based on MSE and R-value, respectively. The ResNet50-GRU-2LSTM model has the best performance on MSE and R-value in all models. The reason might be using the characteristics of ResNet, such as solid feature extraction ability and identity mapping that assist in tackling the vanishing gradient problem. However, the median R-value of the experimental results shows that (see Fig. 16) the ResNet50+LSTM performance can be competitive.

As the hybrid forecasting models consist of several convolutional and recurrent layers and they have a large number of settings (hyper-parameters), we proposed an effective optimiser (multi-strategy differential covariance matrix evolutionary algorithm (ADCMA)) to tune these hyper-parameters and improve the total effectiveness of the hybrid model. Based on the previous findings, the best-performed model was combined with four modern and prosperous meta-heuristics, including DE, AOA, CMA-ES, GNDO and ADCMA, and then compared with ADCMA. The forecasting errors of five hybrid models were tested with and without an optimiser. The statistical test results (MSE and R-value in Fig. 17) indicated that the ADCMA-ResNet50-GRU-2LSTM test results consistently surpassed the 16% effectiveness level compared with the hybrid model with pre-defined hyper-parameters. Furthermore, Fig. 17 demonstrates the statistical consequences with the corresponding MSE and R values and discloses the proposed forecasting hybrid model (ADCMA-ResNet50-GRU-2LSTM) achieved a significantly superior performance compared to the other different hybrid models.

When the number of hyper-parameters increases, traditional tuners, such as the grid or deterministic search algorithms, are inefficient due to the extensive computational cost. Meta-heuristic algorithms have been applied in a considerable number of recent studies to optimise the performance of complex and advanced deep learning models [65,66]. Fig. 18 shows the convergence rate of five hyper-parameter optimisers combined with the ResNet50-GRU-2LSTM model. The ten hyper-parameters involved were learning rate, batch size, filter number and

Table 6
The average SRAD prediction accuracy of ten independent runs for five popular machine learning models, three stacked recurrent models and four hybrid deep learning models. The batch size and learning rate are 254 and 10^{-5}, respectively.

	5CNN_3LSTM					5CNN_3GRU					5CNN_3BiLSTM				
	MSE	RMSE	R-value	MAE	SMAPE	MSE	RMSE	R-value	MAE	SMAPE	MSE	RMSE	R-value	MAE	SMAPE
Mean	0.0156	0.1234	0.7511	0.0874	0.1296	0.0161	0.1250	0.7438	0.0883	0.1306	0.0158	0.1240	0.7433	0.0880	0.1298
Min	0.0146	0.1188	0.6945	0.0833	0.1215	0.0146	0.1184	0.6836	0.0823	0.1238	0.0146	0.1177	0.6929	0.0834	0.1240
Max	0.0172	0.1300	0.7885	0.0923	0.1359	0.0212	0.1425	0.7961	0.1026	0.1397	0.0193	0.1373	0.7955	0.0981	0.1366
Median	0.0157	0.1238	0.7575	0.0880	0.1298	0.0156	0.1235	0.7427	0.0873	0.1301	0.0155	0.1230	0.7493	0.0873	0.1296
STD	0.0008	0.0036	0.0327	0.0030	0.0043	0.0019	0.0071	0.0415	0.0057	0.0056	0.0014	0.0057	0.0376	0.0042	0.0048
	5CNN_GRU-2LSTM					5CNN_2GRU-LSTM					4CNN_GRU-2LSTM				
	MSE	RMSE	R-value	MAE	SMAPE	MSE	RMSE	R-value	MAE	SMAPE	MSE	RMSE	R-value	MAE	SMAPE
Mean	0.0154	0.1223	0.7635	0.0872	0.1297	0.0159	0.1247	0.7609	0.0880	0.1283	0.0163	0.1256	0.7521	0.0890	0.1306
Min	0.0116	0.1044	0.6859	0.0795	0.1216	0.0147	0.1184	0.7089	0.0835	0.1180	0.0146	0.1183	0.7139	0.0828	0.1172
Max	0.0175	0.1310	0.8787	0.0930	0.1410	0.0176	0.1318	0.7947	0.0932	0.1383	0.0224	0.1459	0.7911	0.1052	0.1388
Median	0.0155	0.1234	0.7735	0.0873	0.1290	0.0162	0.1264	0.7868	0.0893	0.1255	0.0157	0.1242	0.7600	0.0884	0.1284
STD	0.0016	0.0074	0.0558	0.0036	0.0055	0.0010	0.0048	0.0381	0.0035	0.0072	0.0026	0.0092	0.0266	0.0074	0.0074
	3CNN_GRU-2LSTM					2CNN_GRU-2LSTM					CNN_GRU-2LSTM				
	MSE	RMSE	R-value	MAE	SMAPE	MSE	RMSE	R-value	MAE	SMAPE	MSE	RMSE	R-value	MAE	SMAPE
Mean	0.0163	0.1263	0.7760	0.0892	0.1271	0.0169	0.1276	0.7329	0.0916	0.1324	0.0362	0.1880	0.3194	0.1537	0.1926
Min	0.0145	0.1186	0.6984	0.0829	0.1182	0.0146	0.1186	0.6923	0.0834	0.1179	0.0338	0.1798	0.2043	0.1477	0.1818
Max	0.0176	0.1319	0.7966	0.0934	0.1365	0.0227	0.1467	0.7976	0.1061	0.1397	0.0407	0.2003	0.4152	0.1637	0.2027
Median	0.0166	0.1278	0.7875	0.0901	0.1257	0.0153	0.1220	0.7163	0.0870	0.1345	0.0360	0.1886	0.3498	0.1518	0.1902
STD	0.0010	0.0044	0.0315	0.0034	0.0056	0.0031	0.0108	0.0389	0.0092	0.0073	0.0023	0.0071	0.0839	0.0054	0.0081
	Xception_LSTM					Resnet50_LSTM					Resnet50_GRU_2LSTM				
	MSE	RMSE	R-value	MAE	SMAPE	MSE	RMSE	R-value	MAE	SMAPE	MSE	RMSE	R-value	MAE	SMAPE
Mean	0.0145	0.1191	0.7754	0.0812	0.1226	0.0143	0.1185	0.7765	0.0810	0.1219	**0.0137**	0.1150	**0.7837**	0.0782	0.1228
Min	0.0138	0.1152	0.7113	0.0722	0.1060	0.0130	0.1138	0.7119	0.0744	0.1129	0.0103	0.0956	0.7313	0.0605	0.1161
Max	0.0163	0.1269	0.8349	0.0871	0.1328	0.0153	0.1232	0.8463	0.0836	0.1304	0.0146	0.1204	0.8771	0.0825	0.1275
Median	0.0143	0.1188	0.7625	0.0813	0.1250	0.0145	0.1186	0.8023	0.0819	0.1222	0.0142	0.1176	0.7723	0.0809	0.1235
STD	0.0008	0.0035	0.0493	0.0040	0.0108	0.0006	0.0030	0.0496	0.0028	0.0068	0.0013	0.0077	0.0478	0.0069	0.0046

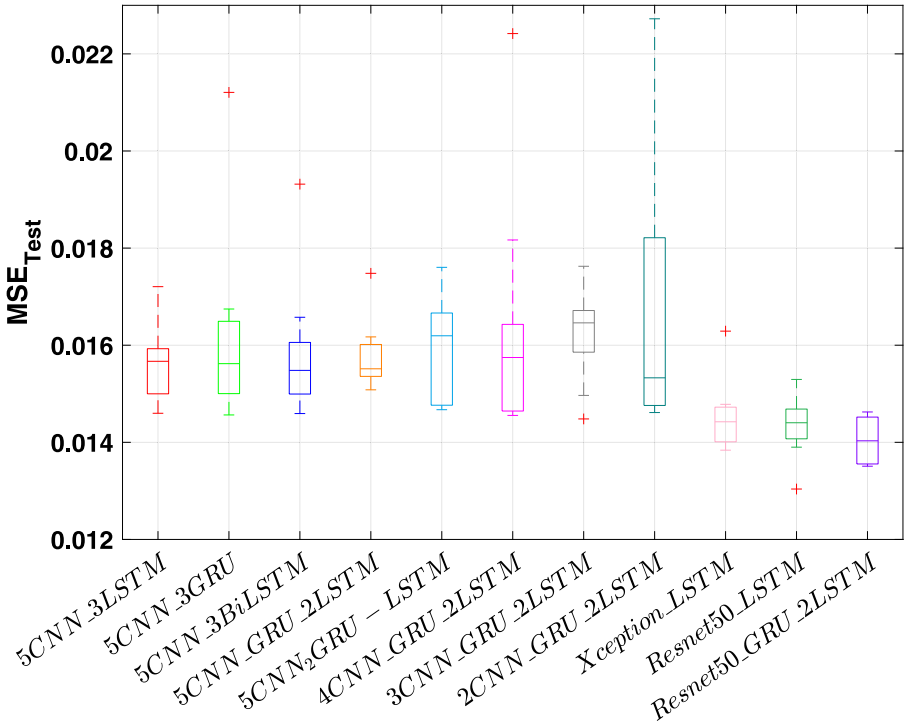

Fig. 15. The average MSE of the validation of 11 hybrid convolutional learning models for predicting solar radiation. 10-fold cross-validation was used to evaluate the performance of the models.

size in convolutional layers, method of weights initialiser, coefficient of dropout layers, hidden size in GRU and LSTM layers, and solver methods. In this figure, ADCMA convergence greater than in other optimisers at the beginning of the search and remains so until the end. Furthermore, the proposed hybrid algorithm shows a considerable performance when it comes to tuning a large number of hyperparameters of deep learning models compared with the standard DE and CMA-ES. From Fig. 18, It can perform a global optimisation of the hyperparameter space computationally efficiently, can adapt to the problem at hand, handle non-differentiable objectives, and is robust to

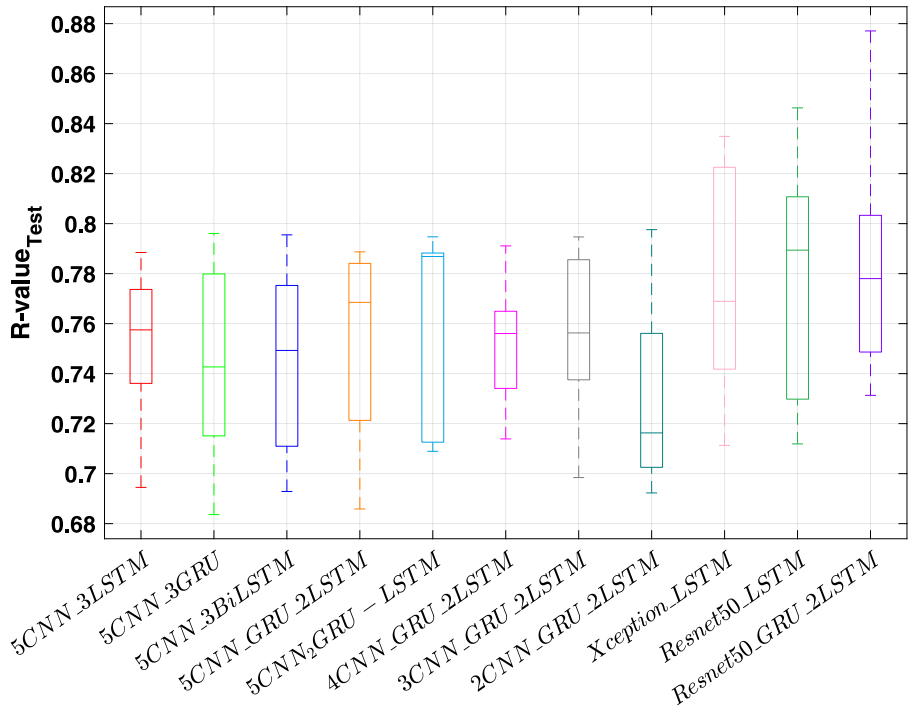

Fig. 16. The average R-value of the validation of 11 hybrid convolutional learning models for predicting solar radiation.

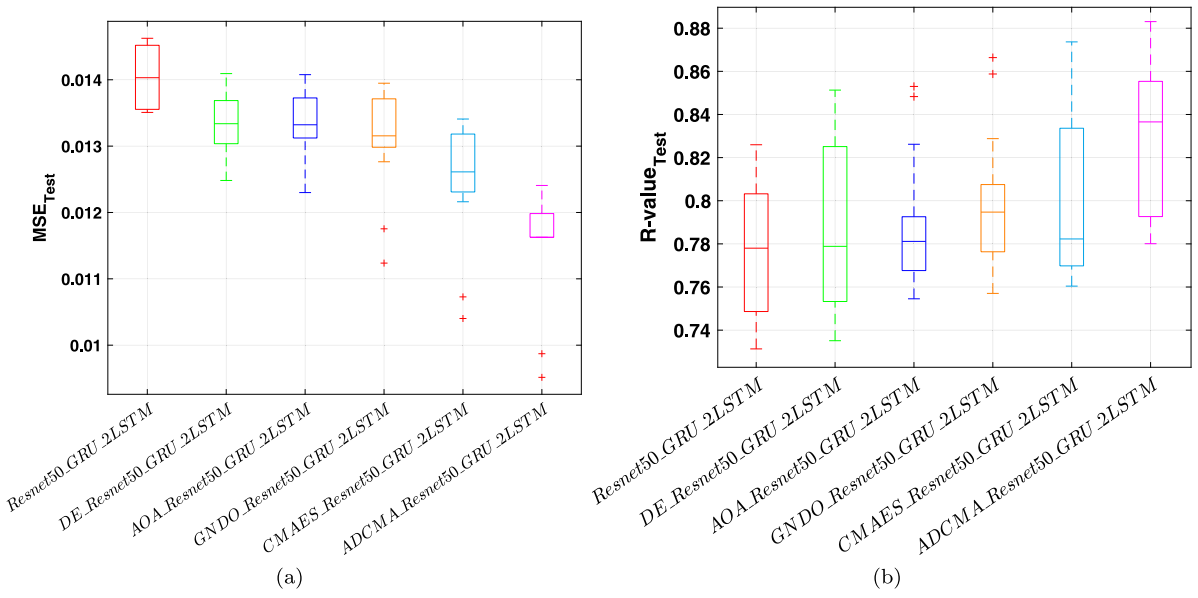

Fig. 17. The statistical performance of five hybrid ResNet50 models with hyper-parameters tuner and without tuner, (a) MSE validation, (b) R-value metric for ten independent training runs with 10-fold cross-validation.

noise. These advantages can lead to better performance of the deep learning model and a more efficient search of the hyperparameter space. Meanwhile, the performance of CMA-ES is competitive compared with DE, AOA, and GNDO methods.

The line chart of the solar radiation forecasting results with the actual value in testing sets in the case study is shown in Fig. 19. This figure shows that the nonlinear and non-stationary solar radiation levels are high due to meteorological conditions. Furthermore, We are comparing the performance of 11 different ML models on forecasting the solar radiation task and using the distance between the predicted and true data as the evaluation metric. We have trained each model on the same dataset and recorded the accuracy for each model on the test set over time. We have plotted the results in a line chart, with the x-axis representing time and the y-axis representing predicted values. When analysing the line chart (Fig. 19), we observe the following trends and patterns: First, Model Resnet50-GRU-2LSTM consistently outperforms the other models over time, increasing accuracy from around 80% to over 90% by the end of the period (can be seen in Fig. 17). Second,

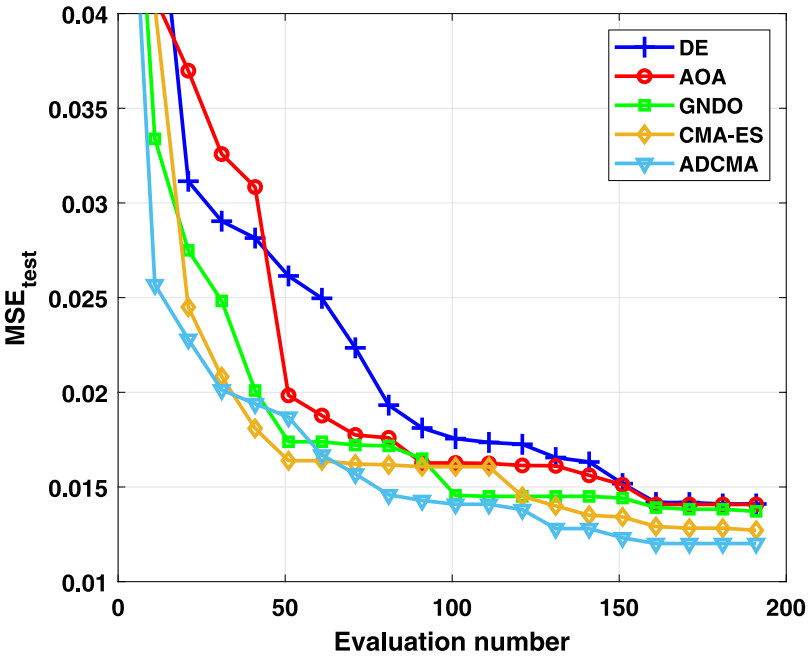

Fig. 18. A convergence rate comparison for four popular population-based optimisation algorithms and the proposed optimiser (ADCMA) in order to find the optimal ten hyper-parameters of the hybrid residual deep learning model (Resnet50-GRU-2LSTM).

model 5CNN-3LSTM shows a fluctuating trend, with various accuracy over time. There are a few sudden drops in accuracy around the halfway point, but it recovers by the end of the period. Third, based on these trends and patterns, we can observe that most recurrent models perform acceptably if the fluctuation rate keeps narrowing. We can also identify the factors contributing to the proposed model's superior performance. For example, we might investigate whether the hybrid model has a larger capacity or is better able to handle the complexity of the dataset. According to the comprehensive evaluation of the proposed hybrid models, it can be sufficiently deduced that the proposed solar radiation forecasting framework retains more heightened prediction accuracy than other models and offers significantly lower bias.

4.3. Sensitivity analysis

We developed a sensitivity analysis to investigate the hidden relationships between hyper-parameters. As sensitivity analysis specifies how various hyper-parameters affect the total performance when forecasting neural models based on validating all feasible combinations of the parameters. The interaction between learning rate and batch size can be seen in Fig. 20(a). This MSE landscape of learning rate and batch size is multi-modal and non-convex, and the dark blue areas indicate the best configuration; for instance, an area between learning rate= $[10^{-4}, 10^{-5}]$, and batch size=$[64, 128]$. Fig. 20(b) depicts a complex relationship between the number of filters and filter size in convolutional layers. A small number of filters (<30) with small filter sizes (<5) is clearly visible. Selecting a suitable technique to initialise the layers' weights can be challenging when combined with other hyper-parameters. In Fig. 20(c), there is no direct relationship between initialiser methods and the dropout layer coefficient, so this investigation will remain open for future study. The 'he' and 'glorot' (also known as 'Xavier') initialisers performed best, and lower dropout values can be better options to reduce learning errors. The most interesting findings of this sensitivity analysis are the MSE landscape of the GRU and the size of LSTM, which is hidden due to complexity and multi-modality (see Figs. 20(d) and (e)). Last but not least, both solvers, 'rmsprop' and 'adam', performed better than 'sgdm' (Fig. 20(f)).

5. Conclusions

Forecasting short-term solar radiation is challenging because of the intermittent, chaotic nature of solar radiation and atmospheric situations. This article proposed a new hybrid deep learning framework to predict short-term (one-hour ahead) solar irradiance. This framework was used for real data acquired from the National Data Buoy Center, Station OKSI2 - Oak St., Chicago. A detailed pre-processing analysis was applied to detect and clean anomalies, and then the data were normalised. In order to improve the prediction of solar radiation, an autoencoder was used to detect and remove outliers. As the decomposition method can effectively reduce the projection error of the hybrid learning network, we developed a reliable and adaptive decomposition method consisting of a multivariate empirical mode decomposition algorithm and 1+1EA-Nelder–Mead simplex direct search. Thus, the AMEMD method is much more effective than the standard EMD method. A fast and effective optimisation algorithm, including four differential mutation strategies and CMA-ES, was also introduced to tune the hyper-parameters of the hybrid learning models. Therefore, the proposed hybrid deep model has a high potential in big data applications and in forecasting other types of renewable energy time-series.

The significant findings of this study are as follows:

1. The proposed hybrid residual learning model (ADCMA-ResNet50-GRU-2LSTM) based on deep residual learning and gated long short-term memory recurrent network facilitated by a fast and effective differential covariance matrix adaptation evolution strategy (ADCMA) performed considerably better in forecasting solar radiation one hour ahead compared with other modern machine learning and hybrid deep learning models. Furthermore, the experimental results showed that the proposed model has outstanding prediction capability and adaptability in non-stationarity solar radiation time-series forecasting.
2. Initialising the control parameters of the MEMD algorithm is challenging because they depend on the time-series data's characteristics. We proposed a fast and robust evolutionary algorithm

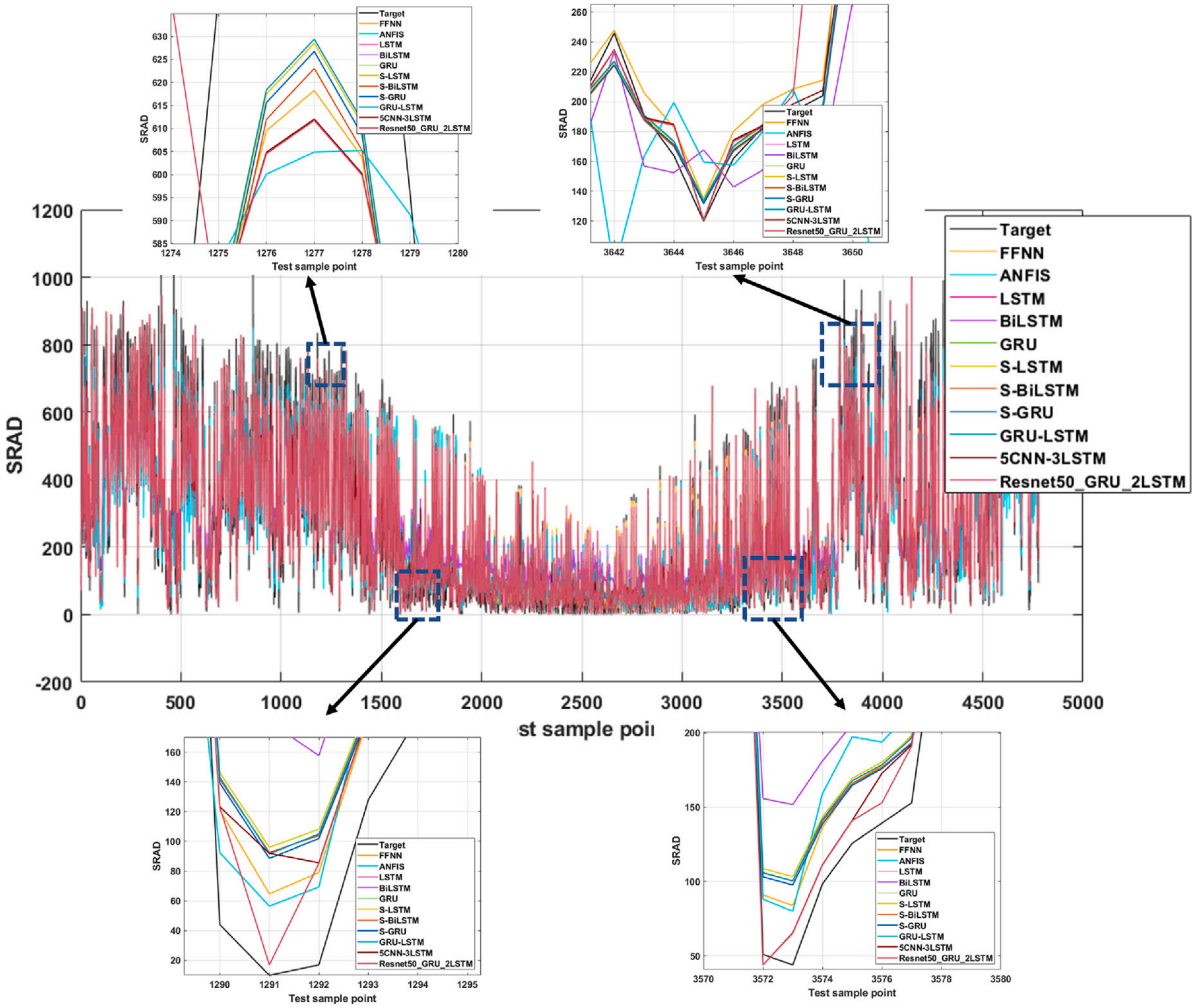

Fig. 19. The prediction of solar radiation using 11 forecasting models.

mixed with a popular local search (Nelder–Mead simplex direct search) to tune the MEMD parameters. The optimal parameters improved the total prediction accuracy by 3%.

3. Tuning the hyper-parameters of the complex hybrid AI-based model significantly increases prediction accuracy and reduces bias. In this study, we suggested an effective optimisation method (multi-strategy differential covariance matrix adaption method- (ADCMA)) to tune ten hyper-parameters of the proposed model. We demonstrated how a sufficient hyper-parameter optimiser could enhance the total performance of the proposed hybrid model by 8%.
4. Moreover, in solar radiation forecasting, the total performance of hybrid convolutional-RNN and stacked RNN models was better than RNN models such as LSTM, BiLSTM and GRU.

To develop the current study as a prospective plan, researchers should consider using state-of-the-art adaptation techniques such as transfer learning and domain adaptation to establish a more robust forecasting model against the domain shift or distributional shift of collected samples from heterogeneous sources. Furthermore, considering evolutionary diversity optimisation [67] methods in order to tune the hyper-parameters of the hybrid forecasting models may have several potential benefits, including improving exploration of solution space, enhancing solution quality by maintaining a diverse set of solutions, increasing robustness and adaptability. Diverse solar radiation datasets should be considered to assess the proposed model's generalisation ability.

CRediT authorship contribution statement

Mehdi Neshat: Conceptualization, Methodology, Software, Validation, Data curation, Writing – original draft, Writing – review & editing, Visualization. **Meysam Majidi Nezhad:** Investigation, Conceptualization, Resources, Data curation, Writing – original draft. **Seyedali Mirjalili:** Investigation, Conceptualization, Supervision, Writing – review & editing. **Davide Astiaso Garcia:** Supervision, Writing – review & editing. **Erik Dahlquist:** Supervision, Writing – review & editing. **Amir H. Gandomi:** Investigation, Conceptualization, Writing – review & editing.

Declaration of competing interest

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

Data availability

Data will be made available on request.

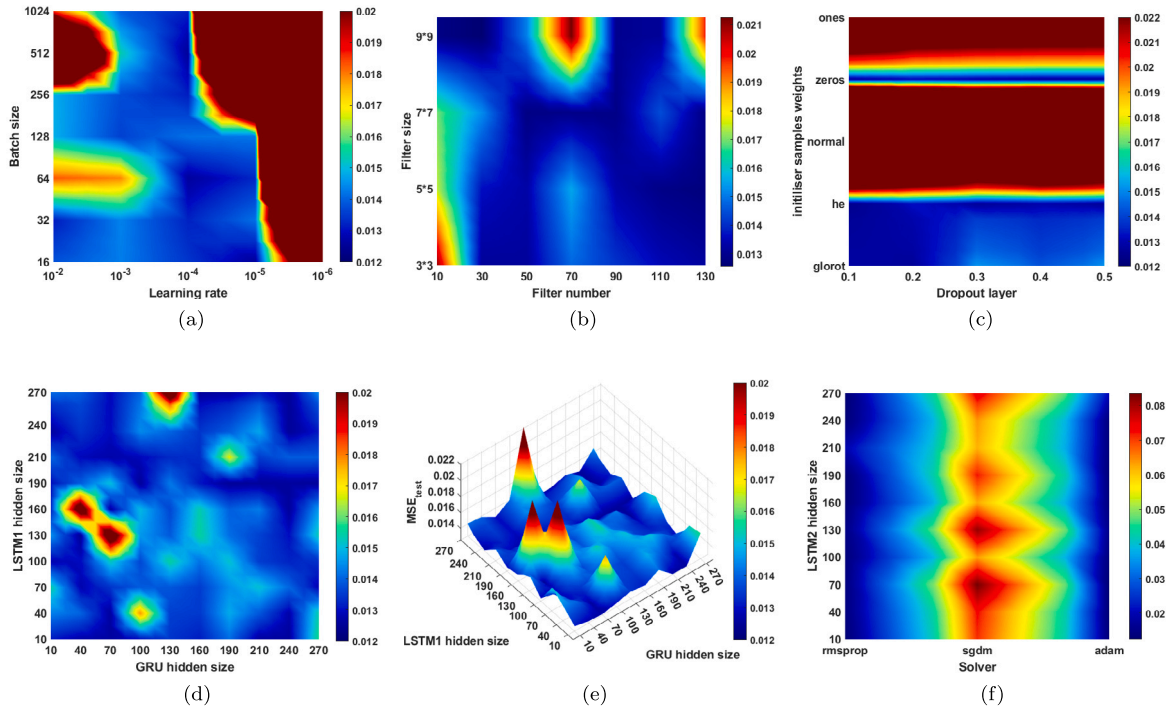

Fig. 20. Ten hyper-parameters sensitivity analysis landscape of the best-found configuration using ADCMA. Dark blue and red show the minimum and maximum MSE validation.

References

[1] Güney T. Solar energy, governance and CO2 emissions. Renew Energy 2022;184:791–8.

[2] Jiang Y, Long H, Zhang Z, Song Z. Day-ahead prediction of bihourly solar radiance with a Markov switch approach. IEEE Trans Sustain Energy 2017;8(4):1536–47.

[3] Ghimire S, Deo RC, Casillas-Pérez D, Salcedo-Sanz S. Improved complete ensemble empirical mode decomposition with adaptive noise deep residual model for short-term multi-step solar radiation prediction. Renew Energy 2022;190:408–24.

[4] Kazantzidis A, Nikitidou E, Salamalikis V, Tzoumanikas P, Zagouras A. New challenges in solar energy resource and forecasting in Greece. Int J Sustain Energy 2018;37(5):428–35.

[5] Ağbulut Ü, Gürel AE, Biçen Y. Prediction of daily global solar radiation using different machine learning algorithms: Evaluation and comparison. Renew Sustain Energy Rev 2021;135:110114.

[6] Bayrakçı HC, Demircan C, Keçebaş A. The development of empirical models for estimating global solar radiation on horizontal surface: A case study. Renew Sustain Energy Rev 2018;81:2771–82.

[7] Fan J, Wang X, Wu L, Zhang F, Bai H, Lu X, et al. New combined models for estimating daily global solar radiation based on sunshine duration in humid regions: a case study in South China. Energy Convers Manage 2018;156:618–25.

[8] Zhang X, Li Y, Lu S, Hamann HF, Hodge B-M, Lehman B. A solar time based analog ensemble method for regional solar power forecasting. IEEE Trans Sustain Energy 2018;10(1):268–79.

[9] Hassan MA, Bailek N, Bouchouicha K, Nwokolo SC. Ultra-short-term exogenous forecasting of photovoltaic power production using genetically optimized non-linear auto-regressive recurrent neural networks. Renew Energy 2021;171:191–209.

[10] Shakya A, Michael S, Saunders C, Armstrong D, Pandey P, Chalise S, et al. Solar irradiance forecasting in remote microgrids using Markov switching model. IEEE Trans Sustain Energy 2016;8(3):895–905.

[11] Lauret P, Voyant C, Soubdhan T, David M, Poggi P. A benchmarking of machine learning techniques for solar radiation forecasting in an insular context. Sol Energy 2015;112:446–57.

[12] Wang H, Liu Y, Zhou B, Li C, Cao G, Voropai N, et al. Taxonomy research of artificial intelligence for deterministic solar power forecasting. Energy Convers Manage 2020;214:112909.

[13] Liu Y, Zhou Y, Chen Y, Wang D, Wang Y, Zhu Y. Comparison of support vector machine and copula-based nonlinear quantile regression for estimating the daily diffuse solar radiation: A case study in China. Renew Energy 2020;146:1101–12.

[14] Japkowicz N. Supervised versus unsupervised binary-learning by feedforward neural networks. Mach Learn 2001;42(1):97–122.

[15] Kumari P, Toshniwal D. Deep learning models for solar irradiance forecasting: A comprehensive review. J Clean Prod 2021;318:128566.

[16] Zendehboudi A, Baseer MA, Saidur R. Application of support vector machine models for forecasting solar and wind energy resources: A review. J Clean Prod 2018;199:272–85.

[17] Alizamir M, Kim S, Kisi O, Zounemat-Kermani M. A comparative study of several machine learning based non-linear regression methods in estimating solar radiation: Case studies of the USA and Turkey regions. Energy 2020;197:117239.

[18] Qing X, Niu Y. Hourly day-ahead solar irradiance prediction using weather forecasts by LSTM. Energy 2018;148:461–8.

[19] Peng T, Zhang C, Zhou J, Nazir MS. An integrated framework of bi-directional long-short term memory (BiLSTM) based on sine cosine algorithm for hourly solar radiation forecasting. Energy 2021;221:119887.

[20] Sangrody H, Zhou N, Tutun S, Khorramdel B, Motalleb M, Sarailoo M. Long term forecasting using machine learning methods. In: 2018 IEEE power and energy conference at illinois. IEEE; 2018, p. 1–5.

[21] Jiang Y. Computation of monthly mean daily global solar radiation in China using artificial neural networks and comparison with other empirical models. Energy 2009;34(9):1276–83.

[22] Voyant C, Notton G, Darras C, Fouilloy A, Motte F. Uncertainties in global radiation time series forecasting using machine learning: The multilayer perceptron case. Energy 2017;125:248–57.

[23] Linares-Rodriguez A, Ruiz-Arias JA, Pozo-Vazquez D, Tovar-Pescador J. An artificial neural network ensemble model for estimating global solar radiation from meteosat satellite images. Energy 2013;61:636–45.

[24] Kisi O. Modeling solar radiation of mediterranean region in Turkey by using fuzzy genetic approach. Energy 2014;64:429–36.

[25] Işık E, Inallı M. Artificial neural networks and adaptive neuro-fuzzy inference systems approaches to forecast the meteorological data for HVAC: The case of cities for Turkey. Energy 2018;154:7–16.

[26] Kaba K, Sarıgül M, Avcı M, Kandırmaz HM. Estimation of daily global solar radiation using deep learning model. Energy 2018;162:126–35.

[27] Larochelle H, Erhan D, Courville A, Bergstra J, Bengio Y. An empirical evaluation of deep architectures on problems with many factors of variation. In: Proceedings of the 24th international conference on machine learning. 2007, p. 473–80.

[28] Zang H, Cheng L, Ding T, Cheung KW, Wang M, Wei Z, et al. Application of functional deep belief network for estimating daily global solar radiation: A case study in China. Energy 2020;191:116502.

[29] Wang J, Jiang H, Wu Y, Dong Y. Forecasting solar radiation using an optimized hybrid model by cuckoo search algorithm. Energy 2015;81:627–44.

[30] Qian Z, Pei Y, Zareipour H, Chen N. A review and discussion of decomposition-based hybrid models for wind energy forecasting applications. Appl Energy 2019;235:939–53.

[31] Dragomiretskiy K, Zosso D. Variational mode decomposition. IEEE Trans Signal Process 2013;62(3):531–44.

[32] ur Rehman N, Aftab H. Multivariate variational mode decomposition. IEEE Trans Signal Process 2019;67(23):6039–52.

[33] Chen KK, Tu JH, Rowley CW. Variants of dynamic mode decomposition: boundary condition, Koopman, and Fourier analyses. J Nonlinear Sci 2012;22(6):887–915.

[34] Kumar DK, Pah ND, Bradley A. Wavelet analysis of surface electromyography. IEEE Trans Neural Syst Rehabil Eng 2003;11(4):400–6.

[35] Huang NE, Shen Z, Long SR, Wu MC, Shih HH, Zheng Q, et al. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. Proc R Soc Lond Ser A Math Phys Eng Sci 1998;454(1971):903–95.

[36] Huang NE, Wu Z. A review on Hilbert-huang transform: Method and its applications to geophysical studies. Rev Geophys 2008;46(2).

[37] Fan X. A method for the generation of typical meteorological year data using ensemble empirical mode decomposition for different climates of China and performance comparison analysis. Energy 2022;240:122822.

[38] Rehman N, Mandic DP. Multivariate empirical mode decomposition. Proc R Soc Lond Ser A Math Phys Eng Sci 2010;466(2117):1291–302.

[39] Huang Y, Hasan N, Deng C, Bao Y. Multivariate empirical mode decomposition based hybrid model for day-ahead peak load forecasting. Energy 2022;239:122245.

[40] Wang G, Chen X, Qiao F, Wu Z, Huang N. On intrinsic mode function. Adv Adapt Data Anal 2010;2(3).

[41] Li X, Ouyang G, Richards DA. Predictability analysis of absence seizures with permutation entropy. Epilepsy Res 2007;77(1):70–4.

[42] Friedrich T, Kötzing T, Lagodzinski G, Neumann F, Schirneck M. Analysis of the (1+ 1) EA on subclasses of linear functions under uniform and linear constraints. In: Proceedings of the 14th ACM/SIGEVO conference on foundations of genetic algorithms. 2017, p. 45–54.

[43] Chung J, Gulcehre C, Cho K, Bengio Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. 2014, arXiv preprint arXiv:1412.3555.

[44] Zhang G, Liu D. Causal convolutional gated recurrent unit network with multiple decomposition methods for short-term wind speed forecasting. Energy Convers Manage 2020;226:113500.

[45] Chollet F. Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 1251–8.

[46] Hemdan EE-D, Shouman MA, Karar ME. Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in X-ray images. 2020, arXiv preprint arXiv:2003.11055.

[47] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016, p. 770–8.

[48] Wen L, Li X, Gao L. A transfer convolutional neural network for fault diagnosis based on ResNet-50. Neural Comput Appl 2020;32:6111–24.

[49] Storn R, Price K. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 1997;11(4):341–59.

[50] Piotrowski AP. Review of differential evolution population size. Swarm Evol Comput 2017;32:1–24.

[51] Storn R. On the usage of differential evolution for function optimization. In: Biennial conference of the North American fuzzy information processing society. 519, IEEE Berkeley; 1996.

[52] Zhang J, Sanderson AC. JADE: adaptive differential evolution with optional external archive. IEEE Trans Evol Comput 2009;13(5):945–58.

[53] Hansen N, Müller SD, Koumoutsakos P. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). Evol Comput 2003;11(1):1–18.

[54] Neshat M, Alexander B, Simpson A. Covariance matrix adaptation greedy search applied to water distribution system optimization. 2019, arXiv preprint arXiv:1909.04846.

[55] Hansen N. Benchmarking a BI-population CMA-ES on the BBOB-2009 function testbed. In: Proceedings of the 11th annual conference companion on genetic and evolutionary computation conference: late breaking papers. 2009, p. 2389–96.

[56] Zhang Y, Jin Z, Mirjalili S. Generalized normal distribution optimization and its applications in parameter extraction of photovoltaic models. Energy Convers Manage 2020;224:113301.

[57] Abualigah L, Diabat A, Mirjalili S, Abd Elaziz M, Gandomi AH. The arithmetic optimization algorithm. Comput Methods Appl Mech Engrg 2021;376:113609.

[58] Neshat M, Alexander B, Sergiienko NY, Wagner M. Optimisation of large wave farms using a multi-strategy evolutionary framework. In: Proceedings of the 2020 genetic and evolutionary computation conference. 2020, p. 1150–8.

[59] Ros R, Hansen N. A simple modification in CMA-ES achieving linear time and space complexity. In: Parallel problem solving from nature–PPSN X: 10th international conference, dortmund, Germany, September 13-17, 2008. proceedings 10. Springer; 2008, p. 296–305.

[60] Ghosh S, Das S, Roy S, Islam SM, Suganthan PN. A differential covariance matrix adaptation evolutionary algorithm for real parameter optimization. Inform Sci 2012;182(1):199–219.

[61] National Data Buoy Centre. Station OKSI2 - oak st., Chicago, IL. 2022, https://www.ndbc.noaa.gov/station_page.php?station=oksi2. [Accessed 6 June 2022].

[62] Google. Map data@2022, lake Michigan: the location of station OKSI2. 2022.

[63] Zhao L, Li Z, Qu L, Zhang J, Teng B. A hybrid VMD-LSTM/GRU model to predict non-stationary and irregular waves on the east coast of China. Ocean Eng 2023;276:114136.

[64] Neshat M, Nezhad MM, Abbasnejad E, Mirjalili S, Tjernberg LB, Garcia DA, et al. A deep learning-based evolutionary model for short-term wind speed forecasting: A case study of the lillgrund offshore wind farm. Energy Convers Manage 2021;236:114002.

[65] Neshat M, Nezhad MM, Sergiienko NY, Mirjalili S, Piras G, Garcia DA. Wave power forecasting using an effective decomposition-based convolutional bi-directional model with equilibrium nelder-mead optimiser. Energy 2022;256:124623.

[66] Xie Y, Li C, Tang G, Liu F. A novel deep interval prediction model with adaptive interval construction strategy and automatic hyperparameter tuning for wind speed forecasting. Energy 2021;216:119179.

[67] Do A, Guo M, Neumann A, Neumann F. Analysis of evolutionary diversity optimization for permutation problems. ACM Trans Evol Learn 2022;2(3):1–27.