Advanced Ensemble Model for Solar Radiation Forecasting Using Sine Cosine Algorithm and Newton's Laws

EL-SAYED M. EL-KENAWY[1], (Member, IEEE),
SEYEDALI MIRJALILI[2,3], (Senior Member, IEEE),
SHERIF S. M. GHONEIM[4], (Senior Member, IEEE), **MARWA METWALLY EID**[1],
M. EL-SAID[5,6], **ZEESHAN SHAFI KHAN**[7], **AND ABDELHAMEED IBRAHIM**[8], (Member, IEEE)

[1]Department of Communications and Electronics, Delta Higher Institute of Engineering and Technology (DHIET), Mansoura 35111, Egypt
[2]Centre for Artificial Intelligence Research and Optimization, Torrens University Australia, Fortitude Valley, QLD 4006, Australia
[3]Yonsei Frontier Lab, Yonsei University, Seoul 03722, South Korea
[4]Department of Electrical Engineering, College of Engineering, Taif University, Taif 21944, Saudi Arabia
[5]Department of Electrical Engineering, Faculty of Engineering, Mansoura University, Mansoura 35516, Egypt
[6]Delta Higher Institute of Engineering and Technology (DHIET), Mansoura 35111, Egypt
[7]Department of Computer Science, College of Engineering and Computer Science, Mustaqbal University, Buraydah 52547, Saudi Arabia
[8]Department of Computer Engineering and Control Systems, Faculty of Engineering, Mansoura University, Mansoura 35516, Egypt

Corresponding authors: El-Sayed M. El-Kenawy (skenawy@ieee.org) and Abdelhameed Ibrahim (afai79@mans.edu.eg)

ABSTRACT As research in alternate energy sources is growing, solar radiation is catching the eyes of the research community immensely. Since solar energy generation depends on uncontrollable natural variables, without proper forecasting, this energy source cannot be trusted. For this forecasting, the use of machine learning algorithms is one of the best choices. This paper proposed an optimized solar radiation forecasting ensemble model consisting of pre-processing and training ensemble phases. The training ensemble phase works on an advanced sine cosine algorithm (ASCA) using Newton's laws of gravity and motion for objects (agents). ASCA uses sine and cosine functions to update the agent's position/velocity components by considering its mass. The training ensemble model is then developed using the k-nearest neighbors (KNN) regression. The performance of the proposed ensemble model is measured using a dataset from Kaggle (Solar Radiation Prediction, Task from NASA Hackathon). The proposed ASCA algorithm is evaluated in comparison with the Particle Swarm Optimizer (PSO), Whale Optimization Algorithm (WOA), Genetic Algorithm (GA), Grey Wolf Optimizer (GWO), Squirrel Search Algorithm (SSA), Harris Hawks Optimization (HHO), Hybrid Greedy Sine Cosine Algorithm with Differential Evolution (HGSCADE), Hybrid Modified Sine Cosine Algorithm with Cuckoo Search Algorithm (HMSCACSA), Marine Predators Algorithm (MPA), Chimp Optimization Algorithm (ChOA), and Slime Mould Algorithm (SMA). Obtained results of the proposed ensemble model are compared with those of state-of-the-art models, and significant superiority of the proposed ensemble model is confirmed using statistical analysis such as ANOVA and Wilcoxon's rank-sum tests.

INDEX TERMS Solar radiation, meta-heuristics, machine learning, K-nearest neighbor, sine cosine algorithm, ANOVA test, Wilcoxon's rank-sum test.

I. INTRODUCTION

During the last few decades, an increase in the demand for energy resources results in finding new means of generating energy. Solar energy produced through solar radiation is one

The associate editor coordinating the review of this manuscript and approving it for publication was Dwarkadas Pralhaddas Kothari.

of the natural methods that are currently in use at domestic as well as at commercial levels [1]. Since solar energy production depends on uncontrollable environmental variables, the production amount cannot be accurately planned. Inconsistent and unpredictable solar energy generation will cause catastrophic results and hence reduce the probability of dependency on solar energy.

The use of machine learning (ML) algorithms applied on the historical dataset of solar radiations can forecast the solar radiations from five to ten minutes [2], [3] and in some cases up to twenty-four hours. Belmahdi *et al.* used ARMA and ARIMA models for global forecasting of solar radiation. According to Belmahdi, this methodology can forecast solar radiation for up to one, two, or three months [4]. Nearly accurate forecasting is needed for the stable and consistent supply of solar energy [5]. Daily meteorological data can be collected through different radiometric stations at various locations, and then this dataset can be employed in forecasting the solar radiations [6]. To obtain reliable forecasting results, researchers have developed various algorithms, and their extensions [7]. Many authors recommend probabilistic forecasting to have better risk management. By using predictive models, visualization, and evaluation of forecasting results to produce probabilistic forecasting, a framework named ProbCast is introduced by Browell and Gilbert [8]. The ML algorithms use atmospheric variables such as wind, temperature, latitude, atmospheric pressure, etc., for nearly accurate solar radiation forecasting. Data related to all these atmospheric variables should be regularly collected, stored, and analyzed to have reliable forecasting.

Genetic Algorithm (GA) and Neural Network (NN) modeling approaches are used by various researchers for solar radiation's forecasting [9]. In their research, average atmospheric pressure and other previous day's weather-related data predicated by a NN is provided as an input parameter to another NN. It is found from the literature that the NN model is providing more accurate forecasting of solar radiations. GA is more suitable in the survival of the fittest scenarios [10]. The physical model definition describes the physical state and dynamic motion of the atmosphere through mathematical equation [11]. GA-based algorithms are not suitable in these types of physical models [12] and NN methodology requires tons of input data that also sometimes includes few non-relevant parameters [13].

Comparative study of various solar radiation forecasting models based on NN and other ML techniques has been performed [14]. Narvaez *et al.* proposed a methodology that works in two steps. In the first step, it selects the best data source to have better Spatio-temporal resolution, and in the second step deep earning is used for forecasting solar-radiation [15]. Geographical and meteorological variables of a specific location are the key parameters considered for solar radiation's forecasting [16]–[18]. Al-Hajj *et al.* proposed a predictive model based on Dynamic Recurrent Neural Networks (DRNN) with short-term delay units to forecast the daily intensity of solar radiations [19]. The model provided better results than Root Mean Square Error (RMSE) and Mean Bias Error (MBE). Another challenge in collecting global solar radiation data is to deal with typical weather conditions, including rainfall, wind, fog, snow, thunder, humidity, sunshine, etc. Proper installation of solar radiation measuring sensors (Pyranometers) is required for such a data collection, and these sensors can be costly, and many

countries do not have sufficient network resources to obtain this data [20]–[22]. In these situations, it is preferred to develop empirical models that can utilize the meteorological data measured by nearby stations [23], [24].

Sine Cosine Algorithm (SCA) [25] has high exploitation as compared to other meta-heuristics since it uses a single best solution to guide other candidate solutions. This makes SCA an efficient algorithm in terms of memory usage and convergence speed and it leads the algorithm to be applied in various recent applications [26], [27]. Based on Newton's law of gravity and Newton's law of motion, the Gravitational Search Algorithm (GSA) was proposed [28]. Position, inertial, active, and passive gravitational masses are the properties of each agent (object mass) [29]. The solution of a problem is represented through these properties and is determined by a fitness function. According to the No Free Lunch (NFL) theorem [30], two algorithms can be considered equivalent when their performance is averaged across all the possible problems [31]–[33]. There is no single ML algorithm that can perform forecasting in all possible situations and scenarios; therefore, different ML algorithms are needed to cover all forecasting scenarios [34]. Many ML algorithms are already developed and employed [35]–[41].

Our contribution in this paper is to develop, analyze, and compare an optimized ensemble model based on pre-processing and training ensemble model. This Advanced Sine Cosine Algorithm (ASCA) based model is inspired by Newton's law of gravity and motion. Sine cosine functions are used for updating the agent's position/velocity component based on its mass. The training ensemble model depends on the k-nearest neighbors (KNN) regression. A dataset from Kaggle (Solar Radiation Prediction, Task from NASA Hackathon) is used for the experiments. The proposed ASCA algorithm is evaluated in compared with Particle Swarm Optimizer (PSO) [42], [43], Whale Optimization Algorithm (WOA) [44], [45], Genetic Algorithm (GA) [46], Grey Wolf Optimizer (GWO) [47], Squirrel search algorithm (SSA) [48], Harris Hawks Optimization (HHO) [32], [49], Hybrid Greedy Sine Cosine Algorithm with Differential Evolution (HGSCADE) [50], Hybrid Modified Sine Cosine Algorithm with Cuckoo Search Algorithm (HMSCACSA) [51], Marine Predators Algorithm (MPA) [52], Chimp Optimization Algorithm (ChOA) [53], and Slime Mould Algorithm (SMA) [54]. Major contributions of our work are as follow:

- ASCA algorithm is developed to optimize the weights and it is based on sine cosine optimization algorithms
- ASCA based model is inspired by Newton's law of gravity and motion.
- The ensemble model depends on the k-nearest neighbors (KNN) regression.
- The model is optimized for nearly accurate forecasting of solar radiations.
- The model is applied to an authentic dataset related to solar radiations.
- Results are obtained by conducting experiments.

- Results are compared with those of other available models.
- The performance of the proposed ensemble model is confirmed using a one-way analysis of variance (ANOVA) and Wilcoxon's rank-sum tests.

The paper is organized into different sections, in which section II explains the background and basic methods, section III includes the details of the proposed model, experiments, and results are mentioned in section IV. Section V includes discussion and findings of the paper and finally, the conclusion and future work is mentioned in section VI.

II. BACKGROUND

Most of the ML algorithms work on the principles of predicting upcoming results based on historical data. This section will discuss the dataset that we used, the sine cosine algorithm, regression methods, and ensemble learning techniques.

A. DATASET

This work uses meteorological data from the HI-SEAS (Hawai'i Space Exploration Analog and Simulation), a dataset from Kaggle (Solar Radiation Prediction, Task from NASA Hackathon). It is a dataset of weather stations for the four months (September through December 2016) between Mission IV and Mission V [55], [56]. Statistical analysis of different weather parameters of the dataset is shown in Table 1. The dataset contains different meteorological parameters such as radiations, temperature, pressure, etc. It also includes static analysis of the dataset's attributes.

B. SINE COSINE ALGORITHM

Mirjalili in 2016 proposed a Sine Cosine Algorithm (SCA) for the optimization problem [25]. Sine and cosine oscillations functions are used for updating the position of a candidate solution. A set of random variables are used by the SCA algorithm to indicate the direction of the movement, distance, emphasize/deem-emphasize the destination's effect, and to switch between sine and cosine components [57]. SCA's mechanism for updating the position of different solutions is represented by Eq. 1.

$$X_i^{t+1} = \begin{cases} X_i^t + r_l \times \sin(r_2) \times |r_3 P_i^t - X_i^t| & r_4 < 0.5 \\ X_i^t + r_l \times \cos(r_2) \times |r_3 P_i^t - X_i^t| & r_4 \geq 0.5 \end{cases} \quad (1)$$

where X_i^t is the current solution position in the ith dimension while P_i^t is the best solution position in the ith dimension. The parameters r_2, r_3, and r_4 are random values in [0, 1]. It can be concluded from the Eq. 1 that the agent's position is changed based on the best solution position. In the SCA algorithm, to balance between the exploration and the exploitation processes, the parameter of r_1 is updated as follows.

$$r_1 = a - \frac{a \times t}{t_{max}} \quad (2)$$

where t indicates current iteration number, a is a constant, and the total number of iterations are represented by t_{max}.

Algorithm 1 : Pseudo-Code of SCA [25]
1: **Initialization** SCA population $X_i(i = 1, 2, \ldots, n)$, size of population n, total iterations t_{max}, and objective function F_n.
2: **Initialization** SCA parameters r_2, r_3, r_4
3: **Set** $t = 1$
4: **while** $t \leq t_{max}$ **do**
5:     **Calculate** objective function F_n for agents X_i
6:     **Set** P = best agent position
7:     **Update** r_1 by $r_1 = a - \frac{a \times t}{t_{max}}$
8:     **for** $(i = 1 : i < n + 1)$ **do**
9:         **if** $(r_4 < 0.5)$ **then**
10:         **Update** agent position by
            $X_i^{t+1} = X_i^t + r_l \times \sin(r_2) \times |r_3 P_i^t - X_i^t|$
11:         **else**
12:         **Update** agent position by
            $X_i^{t+1} = X_i^t + r_l \times \cos(r_2) \times |r_3 P_i^t - X_i^t|$
13:         **end if**
14:     **end for**
15:     **Set** $t = t + 1$
16: **end while**
17: **Return** best agent P

In the Algorithm (1), the position of the SCA population, with n agents, is randomly initiated. Step (5) of the Algorithm (1) calculates the objective function for every agent to find out the best solution position. The best solution is represented as P in step (6) of the Algorithm (1). In step (7) of the Algorithm (1) r_1 is updated by using the Eq. 2. Steps (8-13) of the Algorithm (1) update the position of different agents by using Eq. 1. The algorithm (1) will continue working until the predefined criteria are met. The best solution P will be updated by exploring and exploiting the around space.

Since SCA uses a single best solution to guide other candidate solutions, therefore, it has high exploitation as compared to other meta-heuristics. It makes SCA an efficient algorithm in terms of memory usage and convergence speed. Performance of the SCA degrades in scenarios where a large number of locally optimal solutions exist. These scenarios motivated us to work on this algorithm to alleviate this drawback.

C. GRAVITATIONAL SEARCH ALGORITHM

Based on Newton's law of gravity and Newton's law of motion, the Gravitational Search Algorithm (GSA) was proposed [28]. Position, inertial, active, and passive gravitational masses are the properties of each agent (object mass). The solution of a problem is represented through these properties and is determined by a fitness function.

Mathematically, the ith agent position can be defined as follows.

$$X_i = \left(x_i^1, \ldots, x_i^d, \ldots, x_i^n \right), \quad i = 1, 2, \ldots, N \quad (3)$$

TABLE 1. The units of solar radiation dataset.

	Temperature	Pressure	Humidity	WindDir	WindSpeed	SunHours	Radiation
Count	32686.0	32686.0	32686.0	32686.0	32686.0	32686.0	32686.0
Mean	0.46225	0.62940	0.70544	0.39849	0.15417	0.36369	0.12875
STD	0.16760	0.14777	0.27358	0.23111	0.08619	0.31507	0.19743
Min	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
25%	0.32432	0.56757	0.50526	0.22825	0.08321	0.06316	0.00008
50%	0.43243	0.64865	0.81053	0.41019	0.13877	0.29474	0.00097
75%	0.56757	0.72973	0.93684	0.49803	0.19432	0.62105	0.22068
Max	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

where X_i^d indicates the ith agent position in the dth dimension. N is the number of agents (masses).

The agent position x_i^d is updated by the following equation

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1), \qquad (4)$$

where the agent velocity v_i^d is updated by

$$v_i^d(t+1) = rand_i.v_i^d(t) + a_i^d(t) \qquad (5)$$

where $rand_i$ indicates a random number in [0, 1]. The agent i acceleration, a_i^d, at time t and in d^{th} direction is calculated by

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \qquad (6)$$

where $M_{ii}(t)$ represents the ith agent inertial mass. $F_i^d(t)$ is the total gravitational force that acts on agent i in a dimension d and can be represents by the following equation

$$F_i^d(t) = \sum_{j=1, j\neq i}^{N} rand_j F_{ij}^d(t) \qquad (7)$$

where $rand_j$ indicates a random number in [0, 1]. The force $F_{ij}^d(t)$ acting on mass i from mass j is calculated as following equation.

$$F_{ij}^d(t) = G(t)\frac{M_{pi}(t) \times M_{ai}(t)}{||X_i(t), X_j(t)||_2 + \varepsilon}(x_j^d(t) - x_i^d(d)) \qquad (8)$$

where M_{aj} indicates the agent j active gravitational mass, M_{pi} indicates the agent i passive gravitational mass, $G(t)$ is the gravitational constant at time t, ε is a constant, and $||X_i(t), X_j(t)||_2$ represents the Euclidean distance between two agents i and j.

Since all candidate solutions are used to update the position of each solution hence GSA algorithm has the advantage of very high exploratory behavior. In every rotation, all solutions can influence each other based on their distances and quality. However, the accuracy rate of this algorithm is often not very high.

D. REGRESSION METHODS

1) MULTILAYER PERCEPTRON (MLP)

Artificial Neural Networks (ANN) follow the principles of the biological nervous system for information processing and communication among the distributed nodes. The Synapse (the connection between neurons) is used to transmit signals
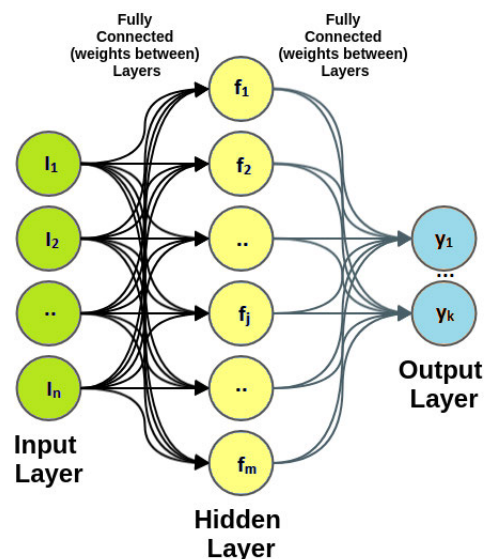

FIGURE 1. Multilayer perceptron (MLP) neural network.

from one neuron to other neurons. Speech recognition, regression, and machine learning algorithms are the most common areas of application of ANN [44], [58]. The learning process and optimization of parameters have a major impact on the performance of ANN. One of the commonly applied ANN is MLP. The MLP structure is shown in Figure 1.

$$S_j = \sum_{i=1}^{n} w_{ij}I_i + \beta_j \qquad (9)$$

where input variable i is represented as I_i. The connection weight between the I_i variable and a hidden layer neuron j is indicated as w_{ij}. β_j represents a layer bias value. By applying the mostly recommended function, sigmoid activation, the output of node j can be calculated as follows.

$$f_j(S_j) = \frac{1}{1 + \exp^{-S_j}} \qquad (10)$$

The following equation defines the output of the network based on the $f_j(S_j)$ value, in Eq. 10, for all neurons in the hidden layer.

$$y_k = \sum_{j=1}^{m} w_{jk}f_j(S_j) + \beta_k \qquad (11)$$

where the weights between a hidden layer neuron j and the output node k is indicated as w_{jk}. The β_k parameter represents the output layer bias value.
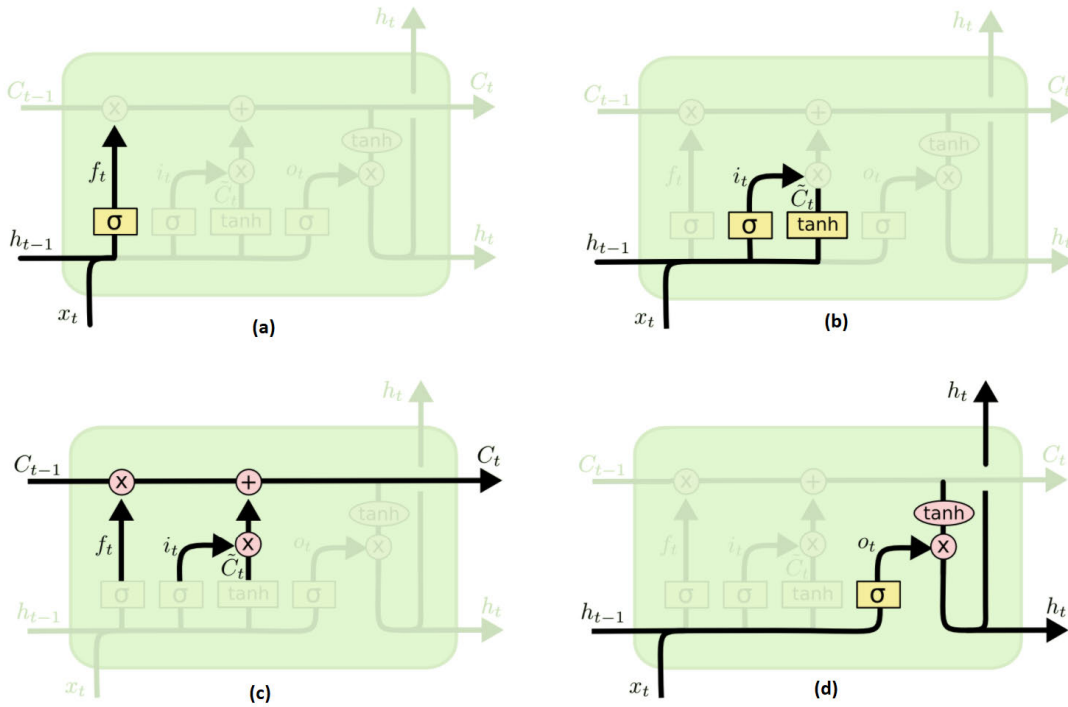
FIGURE 2. LSTM neural network architecture.

2) LONG SHORT TERM MEMORY (LSTM)

LSTM is an improvement over the standard version of ANN and it is a kind of Recurrent Neural Network (RNN) that can be applied to many problems [59]. The main feature of LSTM is to remember the information for a long period and it is more suitable for kind of problems where avoidance of long-term dependency is required. The basic architecture of LSTM is shown in Figure 2.

The first step in LSTM is to decide what information should be discarded from the cell state. A sigmoid layer, named forget gate layer, is used for this as shown in Eq. 12.

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \qquad (12)$$

The next step is about deciding the new information that should be stored in the cell state. The values that need an update are decided by a sigmoid layer named as input gate layer and a new candidate values vector that can be added to the state generated by tanh layer as shown in Eq. 13 and 14.

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \qquad (13)$$
$$C'_t = \tanh(W_i[h_{t-1}, x_t] + b_i) \qquad (14)$$

Using equations 12, 13, and 14, the old cell state C_{t-1} can be updated into a new cell state C_t by the following equation.

$$C_t = f_t \times C_{t-1} + i_t \times C'_t \qquad (15)$$

Output decision based on the cell state is the final step. A sigmoid layer will help to decide about cell state parts that will be moved to output. After that cell state will use tanh

will force values between [-1,1] and will multiply it with the output of the sigmoid gate as mentioned in Eq. 16.

$$h_t = o_t \times \tanh(C_t), \quad o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \qquad (16)$$

3) SUPPORT VECTOR MACHINE

Support vector machines are actually learning models that utilize supervised data and algorithms. Support vector machines are used for classification and regression analysis of data. Hyperplane, in the context of two-dimensional space, represents a line that divides the plane into two subsets [44].

E. ENSEMBLE LEARNING

Ensemble techniques are getting popular in solving the forecasting problem, especially in climate forecasting.

1) AVERAGE ENSEMBLE

In general, the Average Ensemble is one of the simplest ensemble techniques that combines the outputs of base regressors and calculates the mean. This technique aggregates the output of LSTM, SVM, and NN and calculates the mean value as shown in Fig. 3. In this work, the average ensemble is used as a reference ensemble model to evaluate the performance of the proposed ensemble model.

2) KNN ENSEMBLE

KNN is one of the simplest and oldest regressors which is commonly used to decide the regression of unknown instance [60]. KNN utilizes administrative algorithms to test data
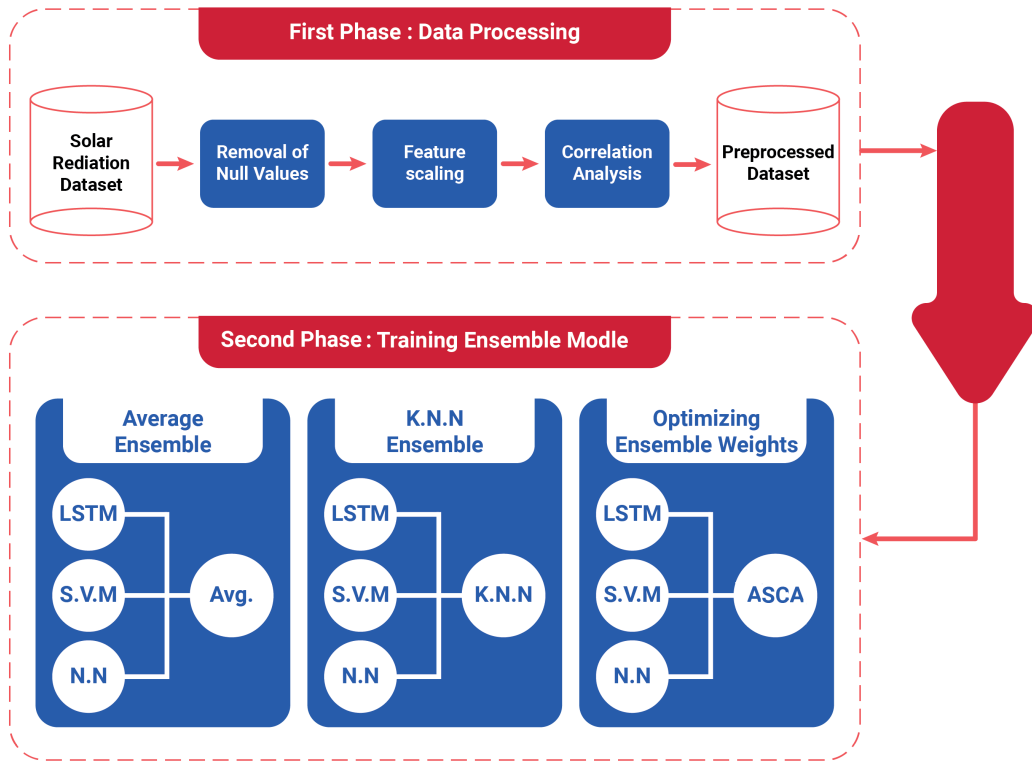
FIGURE 3. Proposed optimized solar radiation forecasting ensemble model.

regression based on the majority class. This algorithm classifies new objects by considering attributes and training data. The target of the KNN algorithm is to discover k objects from the training data that are closer to the testing data then these objects are used for forecasting. KNN strategy is more effective in scenarios where the size of the training data is small. Estimation of a proper value of k is one of the major shortcomings of KNN.

Regression approaches are usually employed to predict the output variables. The regression approach divides the data set into two groups, namely training data and testing data. The testing data is in contrast to training data by using the Euclidean distance. Distance between training data and testing data is calculated and by using heuristics optimal k nearest neighbors are selected. Voting is performed for labels then, with the k nearest multivariate neighbors, an inverse distance weighted average is calculated.

III. PROPOSED MODEL
One of the biggest issues in regression problems is the use of imprecise data that includes many non-relevant features that ultimately increase the error rate of the algorithm. To overcome this problem, we propose a machine learning ensemble model that consists of two phases: data pre-processing phase and training ensemble model as shown in Figure 3.

A. DATA PREPROCESSING
To only count most relevant features, feature analysis is often recommended before regression analysis. So from the dataset,

we removed the rows those contains null values or the rows those have some missing information in order not to have misleading results. Ranges and units of different attributes vary in numbers and since we are using Euclidean distance between two data point so it can affect our algorithm largely. Therefore, we use Min-Max Scalar to bring all the values between 0 and 1 as shown in Eq. 17.

$$I_i' = \frac{I_i - min(I_i)}{max(I_i) - min(I_i)} \tag{17}$$

where I_i' is the scaled value.

To measure the correlation between solar radiation and other attributes, Pearson's correlation is applied in Eq. 18. It helped us to identify strongly correlated and weakly correlated attributes. The correlation coefficient, μ, can be defined as:

$$\mu = \frac{E[i,j] - E[i] * E[j]}{SD_i * SD_j} \tag{18}$$

where SD_i is a variable i standard deviation and SD_j is a variable j standard deviation. $E[i]$ is equal to $\sum i$.

B. ADVANCED SINE COSINE ALGORITHM
The proposed Advanced Sine Cosine Algorithm (ASCA) is inspired by Newton's law of gravity and Newton's law of motion for objects. ASCA is based on sine and cosine functions to update the agent's position for the number of masses as shown in Figures 4 and 5. Based on the agent's mass,
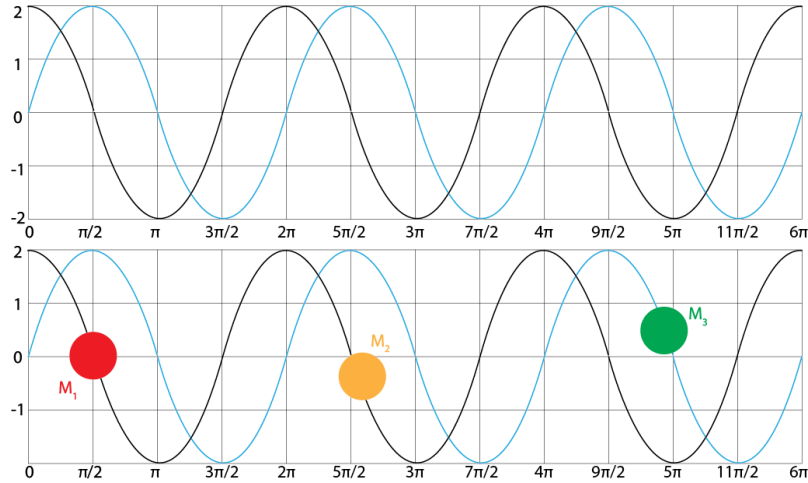
FIGURE 4. Proposed Algorithm: 2D representation of sine-cosine functions with different agents (masses).
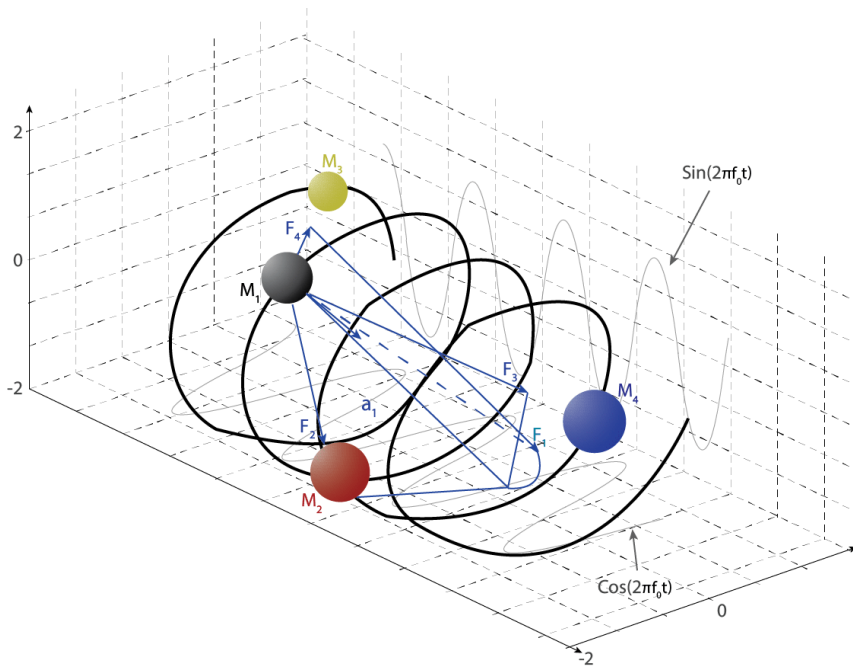
FIGURE 5. Proposed Algorithm: 3D representation of sine-cosine functions with different agents (masses).

the ASCA switches between sine/cosine components and position/velocity component. As we discussed earlier SCA is not very accurate and GSA suffers from low exploration drawback therefore we proposed a hybrid solution by combining the strengths of both these algorithms.

Steps of ASCA are described in Algorithm (2). Mathematically Eq. 19 is used by ASCA to update agent's position for a random parameter that has values in [0, 1] and $rand_{SC} > 0.5$.

$$x_i^d(t+1) = \begin{cases} x_i^d(t) + r_l \times \sin(r_2) \\ \qquad \times |r_3 P_i^d(t) - x_i^d(t)| & r_4 < 0.5 \\ x_i^d(t) + r_l \times \cos(r_2) \\ \qquad \times |r_3 P_i^d(t) - x_i^d(t)| & r_4 \geq 0.5 \end{cases} \quad (19)$$

where $x_i^d(t)$ is the current solution position in the dth dimension while $P_i^d(t)$ is the best solution position in the dth dimension. The parameters r_2, r_3, and r_4 are random values in [0, 1]. As shown in Eq. 19, agent's position is updated based on the best agent position, if $rand_{SC} > 0.5$. In the proposed ASCA algorithm, to balance between the processes of exploitation and exploration, the r_1 parameter is updated as follows.

$$r_1 = a \left(1 - \frac{t}{t_{max}}\right) \quad (20)$$

where t is current iteration number, a indicates a constant, and t_{max} is the total number of iterations.

The agent position x_i^d is updated in the ASCA algorithm by the following equation for $rand_{SC} \leq 0.5$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1), \qquad (21)$$

where the agent velocity v_i^d is updated, as shown in Figure 5, as follows.

$$v_i^d(t+1) = rand_i.v_i^d(t) + a_i^d(t) \qquad (22)$$

where $rand_i$ represents a random number in [0, 1]. The parameter $a_i^d(t)$ represents the agent i acceleration at time t and it is calculated as follows.

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \qquad (23)$$

where $M_{ii}(t)$ represents the ith agent inertial mass. The total gravitational force, in a dimension d, that acts on agent i is represented as $F_i^d(t)$.

The position of the ASCA population, with N agents, is randomly initialized as mentioned in Algorithm (2). Then, the objective function is calculated for each agent to find the position of the best solution. The parameter P indicates the best solution as in Step (5) of the Algorithm (2). r_1 is updated according to Eq. 20 in Step (6). Steps (8-14) update the position of different agents by using Eq. 19 where $rand_{SC} > 0.5$ while steps (17-19) update the position of different agents by using Eq. 21 for $rand_{SC} \leq 0.5$. Steps (3-21) are repeated in the Algorithm (2) until a predefined criterion is met. The best solution P in Step 5 will be updated by exploring and exploiting the around space.

The proposed ASCA algorithm computational complexity is as follow. Time complexity for population of size N with t_{max} iterations can be defined as:

- Initialize ASCA population: $O(1)$.
- Initialize ASCA parameters r_2, r_3, r_4, $rand_i$, $rand_j$, $rand_{SC}$, $t = 1$: $O(1)$.
- Get objective function for agents: $O(t_{max} \times N)$.
- Finding best agent position: $O(t_{max} \times N)$.
- Updating r_1 parameter: $O(t_{max})$.
- Updating agent position: $O(t_{max} \times N)$.
- Updating agent acceleration: $O(t_{max} \times N)$.
- Updating agent velocity: $O(t_{max} \times N)$.
- Updating agent position: $O(t_{max} \times N)$.
- Increasing the iteration number: $O(t_{max})$.
- Producing the best agent: $O(1)$

This analysis shows that the complexity of computations of the proposed ASCA algorithm is $O(t_{max} \times N)$ and $O(t_{max} \times N \times d)$ for the problem with d dimension.

C. TRAINING ENSEMBLE MODEL

Ensemble model instead of picking one best model from the applicants, combines all the models by assigning weight to each model. Ensemble technique is proved as one of the significant methods in improving the prescient capacity of standard models [61]. Ensemble model normally has two phases where in first phase, output variable of the best ensemble

---

Algorithm 2 : Pseudo-Code of the Proposed ASCA

1: **Initialization** ASCA population $X_i = (x_i^1, \ldots, x_i^d, \ldots, x_i^n)$, size of population n, total iterations t_{max}, and objective function F_n.
2: **Initialization** r_2, r_3, r_4, $rand_i$, $rand_j$, $rand_{SC}$, $t = 1$
3: **while** $t \leq t_{max}$ **do**
4:     **Calculate** objective function F_n for each agents x_i^d
5:     **Set** P = best agent position
6:     **Update** r_1 by $r_1 = a\left(1 - \frac{t}{t_{max}}\right)$
7:     **if** $(rand_{SC} > 0.5)$ **then**
8:         **for** $(i = 1 : i < N + 1)$ **do**
9:             **if** $(r_4 < 0.5)$ **then**
10:             **Update** agent position by
    $x_i^d(t+1) = x_i^d(t) + r_l \times \sin(r_2) \times |r_3 P_i^d(t) - x_i^d(t)|$
11:             **else**
12:             **Update** agent position by
    $x_i^d(t+1) = x_i^d(t) + r_l \times \cos(r_2) \times |r_3 P_i^d(t) - x_i^d(t)|$
13:             **end if**
14:         **end for**
15:     **else**
16:         **for** $(i = 1 : i < N + 1)$ **do**
17:             **Update** agent acceleration by
    $a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)}$
18:             **Update** agent velocity by
    $v_i^d(t+1) = rand_i.v_i^d(t) + a_i^d(t)$
19:             **Update** agent position by
    $x_i^d(t+1) = x_i^d(t) + v_i^d(t+1)$
20:         **end for**
21:     **end if**
22:     **Set** $t = t + 1$
23: **end while**
24: **Return** best agent P

---

member is selected in order to obtain the final prediction. Second phase mixes the output variables of ensemble members using a combination algorithm [62].

IV. EXPERIMENTAL RESULTS

Three different scenarios are considered for conducting experiments in order to prove the authenticity of proposed solution. In the first scenario correlation among the input attributes and the solar radiation is analyzed. In the second scenario, base models, LSTM, SVM and NN, are analyzed. Third scenario is based on analysis of ensemble model by using the average ensemble, KNN ensemble and the proposed optimizing ensemble weights model. Meteorological data from the HI-SEAS dataset (Solar Radiation Prediction) [55], [56] is randomly divided into two parts where 80% data is used for training and 20% is used for testing. The dataset has Radiation, Temperature, Pressure and statistical analysis as parameters.

A. PERFORMANCE METRICS

The performance metrics that are used for the experiments include RMSE, Mean Absolute Error (MAE), and MBE [63]. The RMSE metric can be calculated as follow in order to assess the performance:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(H_{p,i} - H_i)^2}{n}} \qquad (24)$$

where $H_{p,i}$ represents a predicted value and H_i indicates the actual measured value. The n parameter represents the total number of values.

The MAE is used to calculate, in a set of predictions, the average amount of errors. It can be calculated as

$$MAE = \frac{\sum_{i=1}^{n}|H_{p,i} - H_i|}{n} \qquad (25)$$

The MBE can show under-predicting or over-predicting state of the tested model. It measures the mean bias of prediction based on the average differences in directions between the predicted and measured values as follows

$$MBE = \frac{\sum_{i=1}^{n}(H_{p,i} - H_i)}{n} \qquad (26)$$

B. FIRST SCENARIO: CORRELATION ANALYSIS

The correlation between input attributes and the solar radiation is presented in Table 2. Prediction of regression is shown in Fig. 6 and the correlation is shown in Fig. 7. It can be observed from Table 2 that wind direction is weakly correlated to solar radiation, therefore, this parameter is decided to be ignored in the experiments. On the other side, the temperature has a higher correlation with solar radiation as compared to that of humidity, wind speed, and sun hours. So in short weakly correlated attributes should not be considered as these can adversely affect the accuracy of ensemble models including our proposed ensemble model.

C. SECOND SCENARIO: BASE MODELS

The second scenario is designed for testing the performance of base models including LSTM, SVM, and NN without involving ensemble techniques. The results of base models using the RMSE, MAE and MBE performance metrics are shown in Table 3. From the results mentioned in Table 3, it can be concluded that the LSTM model, with RMSE of 0.04041579, MAE of 0.03240178, and MBE of -0.00840223, has promising values among the tested base models. However, these results can be improved using the ensemble models.

D. THIRD SCENARIO: ENSEMBLE MODELS

This scenario is based on different ensemble models including average ensemble, KNN ensemble, and the proposed optimizing ensemble weights model. The ensemble models in this experiment utilize the training instances instead of building models to combine/average the results of the three base models (LSTM, SVM, and NN). This can classify unknown
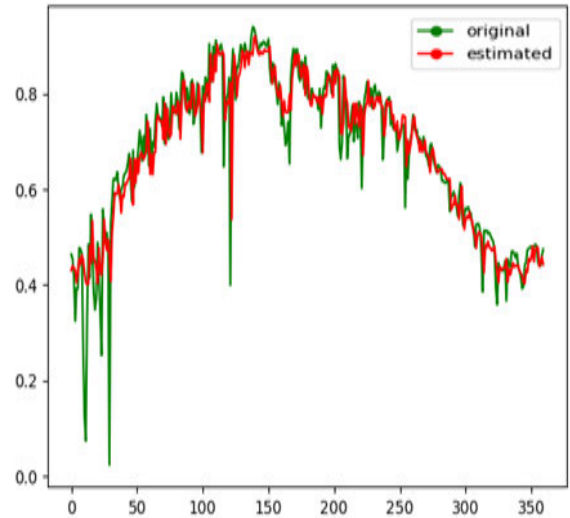

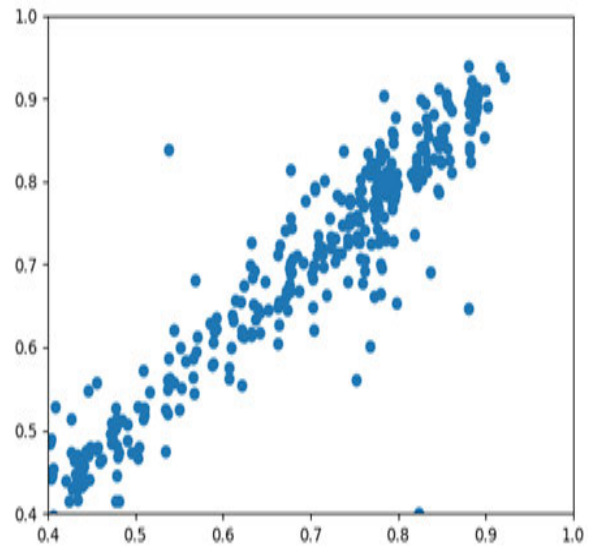**FIGURE 6.** KNN ensemble real predicted values.

FIGURE 7. KNN ensemble correlation.

observations to the regression of the majority and gives the results to predict solar radiation.

Figure 8 shows the optimized weights of the proposed model. Results of different ensemble models are shown in Table 4. From Table 4 it can be seen that the three ensemble models show promising results as compared to those of the base models tested in the second scenario. The proposed ensemble optimizing weights model with RMSE of 0.00175482, MAE of 0.00161235, and MBE of -0.00036521, based on the advanced sine cosine algorithm, gives competitive results compared to the average ensemble and KNN ensemble models. Figure 9 shows the Receiver Operating Characteristics (ROC) curves of the proposed ensemble optimizing weights model versus the average ensemble and KNN ensemble models. The figures show that the proposed model based on the proposed ASCA algorithm is able to distinguish data with a high Area Under the Curve (AUC) with a value of 0.9875.

TABLE 2. Correlation (remove all columns except radiation).

|	Temperature	Pressure	Humidity	WindDir	WindSpeed	SunHours	Radiation
Temperature	1.000000	0.311173	-0.285055	-0.259421	-0.031458	0.355509	0.734955
Pressure	0.311173	1.000000	-0.223973	-0.229010	-0.083639	0.278614	0.119016
Humidity	-0.285055	-0.223973	1.000000	-0.001833	-0.211624	0.087356	-0.226171
WindDir	-0.259421	-0.229010	-0.001833	1.000000	0.073092	-0.129434	-0.230324
WindSpeed	-0.031458	-0.083639	-0.211624	0.073092	1.000000	-0.174944	0.073627
SunHours	0.355509	0.278614	0.087356	-0.129434	-0.174944	1.000000	0.073456
Radiation	0.734955	0.119016	-0.226171	-0.230324	0.073627	0.073456	1.000000
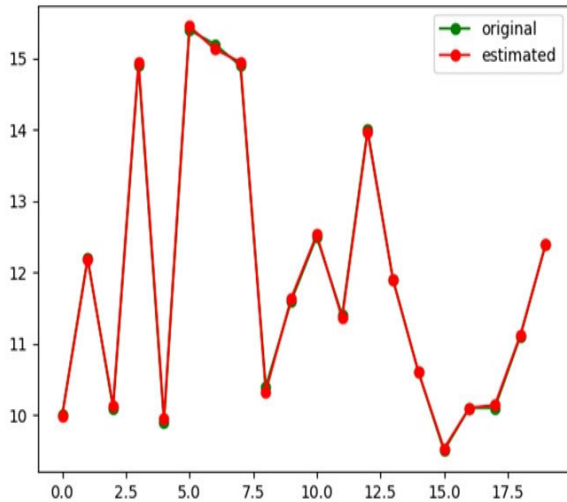
FIGURE 8. Optimized weights predicted values by the proposed model.

ANOVA test is applied to measure the statistical differences between the proposed model and other models that are used for comparison. The hypothesis testing can be formulated using two hypotheses; the null hypothesis (H_0: $\mu_{A1} = \mu_{B1} = \mu_{C1} = \mu_{D1} = \mu_{E1} = \mu_{F1}$), where $A1$: Proposed ensemble weights model, $B1$: KNN ensemble model, $C1$: Average ensemble model, $D1$: SVM model, $E1$: NN model, and $F1$: LSTM model, and the alternate hypothesis (H_1: Means are not all equal). The ANOVA test results are shown in Table 5. Figure 10 shows the ANOVA test results for the proposed and other models based on the objective function. Detailed results obtained through ANOVA test are shown in Table 5. The alternate hypothesis H_1 is preferred based on the results of the test. But to decide the best algorithm another test is needed.

Wilcoxon's rank-sum statistical analysis of the proposed ensemble model in comparison to other models is shown in Table 6. Wilcoxon's rank-sum test will determine if the proposed models and other models' results have a significant difference; p-value < 0.05 will show significant superiority. Hypothesis testing is formulated by two hypotheses; the null hypothesis (H_0: $\mu_{Proposed\ model} = \mu_{KNN\ Ensembl}$, $\mu_{Proposed\ model} = \mu_{Average\ Ensemble}$, $\mu_{Proposed\ model} = \mu_{SVM}$, $\mu_{Proposed\ model} = \mu_{NN}$, and $\mu_{Proposed\ model} = \mu_{LSTM}$) and the alternate hypothesis (H_1: Means are not all equal). The results in Table 6 explain that p-values are less than 0.05 which are achieved between the proposed model and other models. This shows the superiority of the ASCA based proposed model
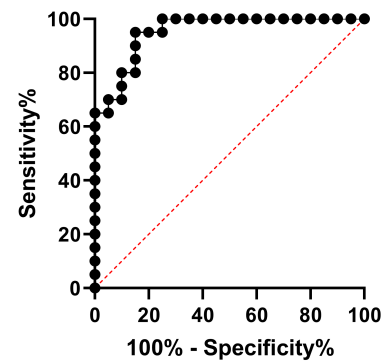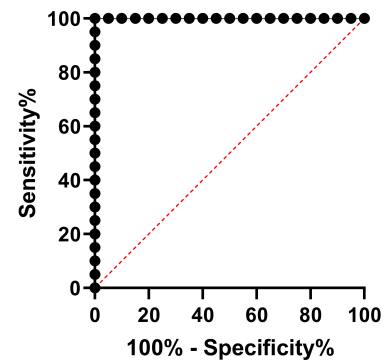

FIGURE 9. ROC curves of the proposed optimizing ensemble model versus average ensemble and KNN ensemble models.

TABLE 3. Experimental results of the base models.

	RMSE	MAE	MBE
LSTM model	0.04041579	0.03240178	-0.00840223
NN model	0.05904678	0.04003057	-0.01351504
SVM model	0.05673949	0.04269841	0.00942546

and also indicates the statistical significance of the algorithm. Thus, the alternate hypothesis H_1 is accepted.

E. FOURTH SCENARIO: ASCA ALGORITHM PERFORMANCE

In this section, the proposed ASCA algorithm is evaluated in compared with PSO [42], [43], WOA [44], [45], GA [46],GWO [47], SSA [48], HHO [32], [49], HGSCADE [50],

TABLE 4. Experimental results of the of the proposed ensemble model compared to other models.

	RMSE	MAE	MBE
Average Ensemble	0.03722151	0.03101721	-0.00386392
KNN Ensemble	0.02368600	0.02835076	-0.00144518
Proposed Optimizing Ensemble Weights	0.00175482	0.00161235	-0.00036521

TABLE 5. ANOVA test results of the proposed ensemble model compared to other models.

	SS	DF	MS	F(DFn, DFd)	P value
Treatment (between columns)	0.04858	5	0.009715	F (5, 120) = 2414	P < 0.0001
Residual (within columns)	0.000483	120	4.03E-06	-	-
Total	0.04906	125	-	-	-

TABLE 6. Wilcoxon's rank-sum statistical results of the proposed ensemble model in comparison to other models.

	LSTM	NN	SVM	Average Ensemble	KNN Ensemble	Optimizing Ensemble
Theoretical median	0	0	0	0	0	0
Actual median	0.04085	0.05964	0.05733	0.03768	0.02418	0.001755
Number of values	21	21	21	21	21	21
Wilcoxon Signed Rank Test						
Signed ranks sum (W)	231	231	231	231	231	231
Positive ranks sum	231	231	231	231	231	231
Negative ranks sum	0	0	0	0	0	0
P value (two tailed)	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001
Estimate or Exact?	Exact	Exact	Exact	Exact	Exact	Exact
P value summary	****	****	****	****	****	****
Significant (alpha=0.05)?	Yes	Yes	Yes	Yes	Yes	Yes
How big is the discrepancy?						
Discrepancy	0.04085	0.05964	0.05733	0.03768	0.02418	0.001755
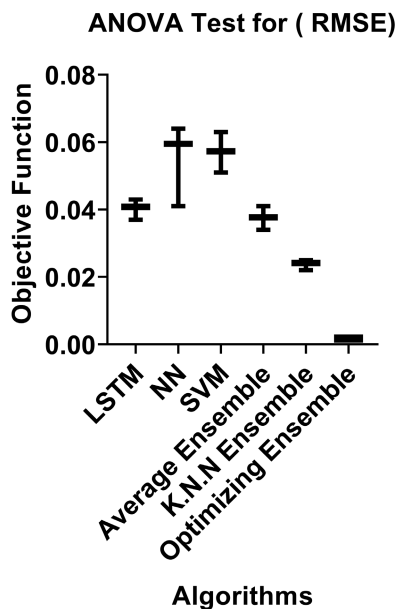
FIGURE 10. ANOVA test for different models.

HMSCACSA [51], MPA [52], ChOA [53], and SMA [54] algorithms. For a fair comparison, the proposed ASCA algorithm and the compared algorithms begin in the experiment with the same number of agents (population) with same size and are applied to the same objective function using same number of iteration, dimensions, and boundaries. Table 7 shows the classification accuracy and the descriptive statistics of the proposed ASCA algorithm compared to other algorithms. The table indicates that the ASCA algorithm achieved better results than compared algorithms.

ANOVA and Wilcoxon's rank-sum tests are performed using 20 runs for a fair comparison between the ASCA algorithm and the compared algorithms. ANOVA test is applied to measure the statistical differences between the proposed algorithm and compared algorithms that are used for comparison as shown in Table 8. The hypothesis testing is formulated here using two hypotheses; the null hypothesis (H_0: $\mu_A = \mu_B = \mu_C = \mu_D = \mu_E = \mu_F = \mu_G = \mu_H = \mu_I = \mu_J = \mu_K = \mu_L$), where A: ASCA algorithm, B: PSO algorithm, C: WOA algorithm, D: GA algorithm, E: GWO algorithm, F: SSA algorithm, G: HHO algorithm, H: HGSCADE algorithm, I: HMSCACSA algorithm, J: MPA algorithm, K: ChOA algorithm, and L: SMA algorithm. The alternate hypothesis (H_1: Means are not all equal). In addition, Wilcoxon's rank-sum statistical analysis of the proposed algorithm in comparison to other algorithms is shown in Table 9. Hypothesis testing is formulated by two hypotheses; the null hypothesis (H_0: $\mu_{ASCA} = \mu_{PSO}$, $\mu_{ASCA} = \mu_{WOA}$, $\mu_{ASCA} = \mu_{GA}$, $\mu_{ASCA} = \mu_{GWO}$, $\mu_{ASCA} = \mu_{SSA}$, $\mu_{ASCA} = \mu_{HHO}$, $\mu_{ASCA} = \mu_{HGSCADE}$, $\mu_{ASCA} = \mu_{HMSCACSA}$, $\mu_{ASCA} = \mu_{MPA}$, $\mu_{ASCA} = \mu_{ChOA}$, $\mu_{ASCA} = \mu_{SMA}$. The alternate hypothesis (H_1: Means are not all equal). This shows the superiority and indicates the statistical significance of

TABLE 7. The classification accuracy and descriptive statistics of the proposed ASCA algorithm compared to other algorithms.

	ASCA	PSO	WOA	GA	GWO	SSA	HHO	HGSCADE	HMSCACSA	MPA	ChOA	SMA
Number of values	20	20	20	20	20	20	20	20	20	20	20	20
Minimum	0.001755	0.008712	0.009914	0.006157	0.004875	0.005742	0.005542	0.002813	0.002459	0.004414	0.006845	0.003237
25% Percentile	0.001755	0.009812	0.01001	0.007157	0.006875	0.007742	0.006542	0.003127	0.003459	0.005414	0.007845	0.004237
Median	0.001755	0.009812	0.01001	0.007157	0.006875	0.007742	0.006542	0.003127	0.003459	0.005414	0.007845	0.004237
75% Percentile	0.001755	0.009812	0.01001	0.007157	0.006875	0.007742	0.006542	0.003127	0.003459	0.005414	0.007845	0.004237
Maximum	0.001755	0.01098	0.0198	0.009216	0.008787	0.008742	0.007542	0.004627	0.004746	0.006414	0.008885	0.006237
Range	0	0.002269	0.009888	0.003059	0.003913	0.003	0.002	0.001814	0.002287	0.002	0.002039	0.003
Mean	0.001755	0.009766	0.0105	0.00721	0.00682	0.007642	0.006535	0.003186	0.003532	0.005445	0.007897	0.004337
Std. Deviation	0	0.000431	0.00219	0.000522	0.000673	0.000553	0.000326	0.0003463	0.0004564	0.000354	0.000399	0.000553
Std. Error of Mean	0	9.64E-05	0.00049	0.000117	0.00015	0.000124	7.29E-05	0.00007743	0.000102	7.91E-05	8.92E-05	0.000124
Coefficient of variation	0.000%	4.416%	20.86%	7.244%	9.865%	7.230%	4.987%	10.87%	12.92%	6.492%	5.054%	12.74%
Geometric mean	0.001755	0.009757	0.01036	0.007193	0.006787	0.00762	0.006527	0.003172	0.003506	0.005434	0.007888	0.004307
Geometric SD factor	1	1.046	1.165	1.07	1.108	1.082	1.052	1.096	1.134	1.068	1.051	1.124
Harmonic mean	0.001755	0.009747	0.01026	0.007179	0.006752	0.007596	0.006519	0.00316	0.00348	0.005423	0.007878	0.00428
Skewness		-0.174	4.471	2.833	-0.1142	-2.164	0.06551	4.133	1.183	0.03066	0.5963	2.164
Kurtosis		5.32	20	13.37	7.077	8.208	9.278	18.21	4.844	6.352	4.955	8.208
Sum	0.0351	0.1953	0.21	0.1442	0.1364	0.1528	0.1307	0.06372	0.07065	0.1089	0.1579	0.08674

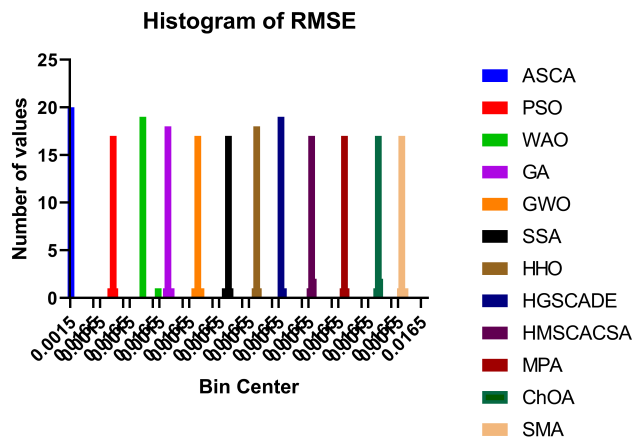**TABLE 8.** ANOVA test results of the proposed ASCA algorithm in comparison to other algorithms.

	SS	DF	MS	F (DFn, DFd)	P value
Treatment (between columns)	0.001553	11	0.0001412	F (11, 228) = 240.8	P < 0.0001
Residual (within columns)	0.0001337	228	5.863E-07	-	-
Total	0.001687	239	-	-	-
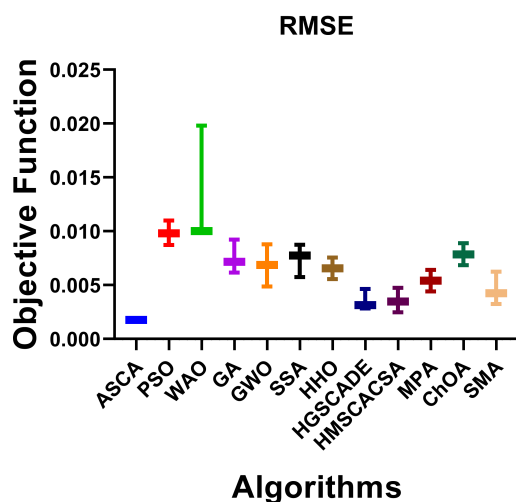
FIGURE 11. Algorithms' performance versus the objective function based on the RMSE parameter.

FIGURE 12. Histogram of RMSE of the presented algorithm compared to other algorithms based on number of values with Bin Center range (0.0015 - 0.0165).

the ASCA algorithm. Thus, the alternate hypothesis H_1 is accepted.

In Figure 11, algorithms' performance versus the objective function is shown based on the RMSE parameter. It can be noted that the minimum, maximum, and average values based on the proposed algorithm are almost the same. This curve indicated the stability of the proposed ASCA algorithm. The histogram of RMSE shown in Figure 12 for different algorithms based on the number of values confirms the stability of the ASCA algorithm. The Residual, Homoscedasticity, quantile-quantile (QQ) plots and heat map

shown in Figure 13, is known as a chance plot. It is mostly used by plotting the quantiles and comparing them to contrast two probability distributions. As the figure shows, the points' distributions in the QQ approximately fit the line. Therefore, the actual and the forecasted residuals were linearly related, thus validating the recommended ASCA efficiency.

V. DISCUSSION

From the results of different experiments that are conducted for evaluating the performance of the proposed solution, it can be seen that the proposed ASCA ensemble weight model outperforms other models in terms of providing accurate forecasting. The algorithms are based on atmospheric vari-

TABLE 9. Wilcoxon's rank-sum statistical results of the proposed ASCA algorithm in comparison to other algorithms.

	ASCA	PSO	WOA	GA	GWO	SSA	HHO	HGSCADE	HMSCACSA	MPA	ChOA	SMA
Theoretical median	0	0	0	0	0	0	0	0	0	0	0	0
Actual median	0.001755	0.009812	0.01001	0.007157	0.006875	0.007742	0.006542	0.003127	0.003459	0.005414	0.007845	0.004237
Number of values	20	20	20	20	20	20	20	20	20	20	20	20
Wilcoxon Signed Rank Test												
Sum of signed ranks (W)	210	210	210	210	210	210	210	210	210	210	210	210
Sum of positive ranks	210	210	210	210	210	210	210	210	210	210	210	210
Sum of negative ranks	0	0	0	0	0	0	0	0	0	0	0	0
P value (two tailed)	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001
Exact or estimate?	Exact	Exact	Exact	Exact	Exact	Exact	Exact	Exact	Exact	Exact	Exact	Exact
P value summary	****	****	****	****	****	****	****	****	****	****	****	****
Significant (alpha=0.05)?	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
How big is the discrepancy?												
Discrepancy	0.001755	0.009812	0.01001	0.007157	0.006875	0.007742	0.006542	0.003127	0.003459	0.005414	0.007845	0.004237
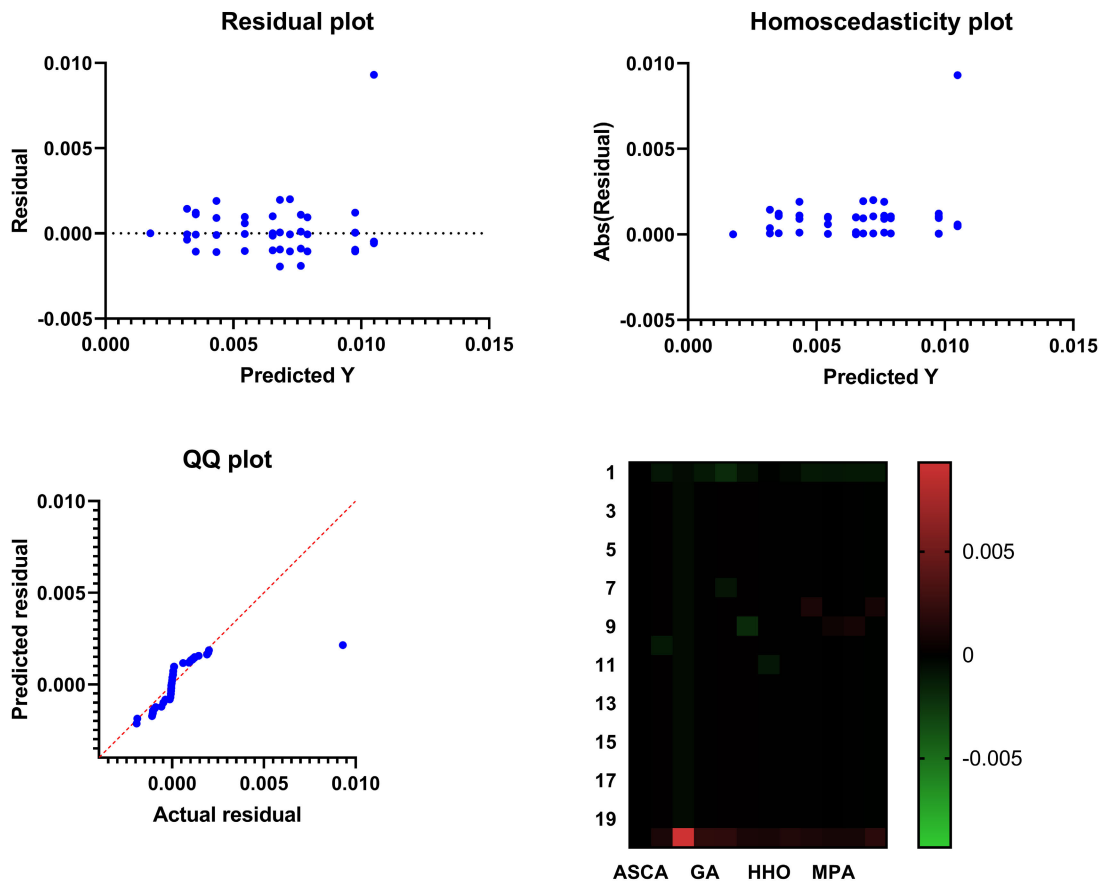
FIGURE 13. Residual, Homoscedasticity, quantile-quantile (QQ) plots and heat map for the proposed and compared algorithms.

ables including temperature, pressure, humidity, etc. These variables should be collected, stored, and analyzed to have more accurate and reliable forecasting. The placement of radiation measuring sensors (pyranometers) is required to collect this kind of data. As constructing such a network is resource-consuming and costly so it is not feasible to have these sensors in every place. Therefore empirical models need to be developed using meteorological data from nearby available stations.

The proposed ensemble optimizing weights model with RMSE of 0.00175482, MAE of 0.00161235, and MBE of -0.00036521, based on the ASCA algorithm, gives com-petitive results compared to the average ensemble and KNN ensemble models. The proposed model based on the proposed ASCA algorithm is able to distinguish data with a high AUC with a value of 0.9875. The analysis based on ANOVA and Wilcoxon's rank-sum tests using more than 20 runs shows the superiority of the ASCA based proposed model and indicates the statistical significance of the algorithm. The classification accuracy and the descriptive statistics of the proposed ASCA algorithm and other algorithms indicate that the proposed algorithm can achieve better results than compared algorithms. ANOVA and Wilcoxon's rank-sum tests show the superiority and indicate the statistical significance of the

ASCA algorithm which validating the recommended ASCA efficiency.

The proposed ASCA algorithm is also tested for the problem of binary classification, but the algorithm shows slow convergence in the experiments. This can be considered as a disadvantage of the ASCA algorithm for such kinds of problems. The proposed algorithm needs some improvements to cover this.

VI. CONCLUSION AND FUTURE DIRECTION

This paper forecasts solar radiation based on a proposed advanced sine cosine algorithm-based ensemble model. The proposed ensemble model shows superiority over the reference base model including LSTM, NN, and SVM. The ASCA based ensemble weights model provided better results over the average ensemble and the KNN ensemble models. Several experiments are conducted and different performance metrics are considered to conclude that the proposed ensemble weights model is best suitable for forecasting solar radiation. The proposed ASCA algorithm is evaluated in comparison with the PSO, WOA, GA, GWO, SSA, HHO, HGSCADE, HMSCACSA, MPA, ChOA, and SMA algorithms. Significant superiority of the proposed ensemble model is also confirmed using statistical analysis such as ANOVA and Wilcoxon's rank-sum tests. In future work, the proposed ASCA algorithm will be applied for other continuous problems, binary problems with a high number of attributes for feature selection and classification problems, and also constrained engineering problems. Different novel optimization techniques to select the best machine learning algorithms and to calculate weights for each can be considered in the future.

ACKNOWLEDGMENT

REFERENCES

[1] J. Li, J. K. Ward, J. Tong, L. Collins, and G. Platt, "Machine learning for solar irradiance forecasting of photovoltaic system," *Renew. Energy*, vol. 90, pp. 542–553, May 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0960148115305747

[2] J. Li, Y. Guo, G. Platt, and J. K. Ward, "Renewable energy aggregation with intelligent battery controller," *Renew. Energy*, vol. 59, pp. 220–228, Nov. 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0960148113001833

[3] H. Zhou, G. Ma, C. Wang, J. Wang, G. Zhang, Y. Tu, and C. Li, "Machine learning methods for solar radiation forecasting: A review," *Renew. Energy*, vol. 105, pp. 569–582, May 2017, doi: 10.1016/j.renene.2016.12.095.

[4] B. Belmahdi, M. Louzazni, and A. E. Bouardi, "One month-ahead forecasting of mean daily global solar radiation using time series models," *Optik*, vol. 219, Oct. 2020, Art. no. 165207, doi: 10.1016/j.ijleo.2020.165207.

[5] R. Ahmed, V. Sreeram, Y. Mishra, and M. D. Arif, "A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization," *Renew. Sustain. Energy Rev.*, vol. 124, May 2020, Art. no. 109792, doi: 10.1016/j.rser.2020.109792.

[6] M. Hossain, S. Mekhilef, M. Danesh, L. Olatomiwa, and S. Shamshirband, "Application of extreme learning machine for short term output power forecasting of three grid-connected PV systems," *J. Cleaner Prod.*, vol. 167, pp. 395–405, Nov. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0959652617317973

[7] S. Sobri, S. Koohi-Kamali, and N. A. Rahim, "Solar photovoltaic generation forecasting methods: A review," *Energy Convers. Manage.*, vol. 156, pp. 459–497, Jan. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0196890417310622

[8] J. Browell and C. Gilbert, "ProbCast: Open-source production, evaluation and visualisation of probabilistic forecasts," in *Proc. Int. Conf. Probabilistic Methods Appl. Power Syst. (PMAPS)*, Aug. 2020, pp. 1–6, doi: 10.1109/pmaps47429.2020.9183441.

[9] S. Quaiyum, S. Rahman, and S. Rahman, "Article: Application of artificial neural network in forecasting solar irradiance and sizing of photovoltaic cell for standalone systems in Bangladesh," *Int. J. Comput. Appl.*, vol. 32, no. 10, pp. 51–56, Oct. 2011.

[10] A. Khosravi, R. N. N. Koury, L. Machado, and J. J. G. Pabon, "Prediction of hourly solar radiation in Abu Musa Island using machine learning algorithms," *J. Cleaner Prod.*, vol. 176, pp. 63–75, Mar. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0959652617330007

[11] A. Khosravi, R. O. Nunes, M. E. H. Assad, and L. Machado, "Comparison of artificial intelligence methods in estimation of daily global solar radiation," *J. Cleaner Prod.*, vol. 194, pp. 342–358, Sep. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0959652618314872

[12] S. Mishra and P. Palanisamy, "Multi-time-horizon solar forecasting using recurrent neural network," in *Proc. IEEE Energy Convers. Congr. Exposit. (ECCE)*, Sep. 2018, pp. 18–24, doi: 10.1109/ecce.2018.8558187.

[13] F. Rodríguez, A. Fleetwood, A. Galarza, and L. Fontán, "Predicting solar energy generation through artificial neural networks using weather forecasts for microgrid control," *Renew. Energy*, vol. 126, pp. 855–864, Oct. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0960148118303793

[14] M. Guermoui, F. Melgani, K. Gairaa, and M. L. Mekhalfi, "A comprehensive review of hybrid models for solar radiation forecasting," *J. Cleaner Prod.*, vol. 258, Jun. 2020, Art. no. 120357, doi: 10.1016/j.jclepro.2020.120357.

[15] G. Narvaez, L. F. Giraldo, M. Bressan, and A. Pantoja, "Machine learning for site-adaptation and solar radiation forecasting," *Renew. Energy*, vol. 167, pp. 333–342, Apr. 2021, doi: 10.1016/j.renene.2020.11.089.

[16] A. K. Yadav and S. S. Chandel, "Solar radiation prediction using artificial neural network techniques: A review," *Renew. Sustain. Energy Rev.*, vol. 33, pp. 772–781, May 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1364032113005959

[17] M. Marzouq, H. E. Fadili, Z. Lakhliai, and K. Zenkouar, "A review of solar radiation prediction using artificial neural networks," in *Proc. Int. Conf. Wireless Technol., Embedded Intell. Syst. (WITS)*, Apr. 2017, pp. 1–6, doi: 10.1109/wits.2017.7934658.

[18] E. Praynlin and J. I. Jensona, "Solar radiation forecasting using artificial neural network," in *Proc. Innov. Power Adv. Comput. Technol. (i-PACT)*, Apr. 2017, pp. 1–7, doi: 10.1109/ipact.2017.8244939.

[19] R. Al-Hajj, A. Assi, and M. M. Fouad, "Forecasting solar radiation strength using machine learning ensemble," in *Proc. 7th Int. Conf. Renew. Energy Res. Appl. (ICRERA)*, Oct. 2018, pp. 184–188, doi: 10.1109/icrera.2018.8567020.

[20] Q. Cao, Y. Liu, K. Lyu, Y. Yu, D. H. W. Li, and L. Yang, "Solar radiation zoning and daily global radiation models for regions with only surface meteorological measurements in China," *Energy Convers. Manage.*, vol. 225, Dec. 2020, Art. no. 113447, doi: 10.1016/j.enconman.2020.113447.

[21] S. Gupta, A. R. Katta, Y. Baldaniya, and R. Kumar, "Hybrid random forest and particle swarm optimization algorithm for solar radiation prediction," in *Proc. IEEE 5th Int. Conf. Comput. Commun. Autom. (ICCCA)*, Oct. 2020, pp. 302–307, doi: 10.1109/iccca49541.2020.9250715.

[22] A. N.-L. Huynh, R. C. Deo, D.-A. An-Vo, M. Ali, N. Raj, and S. Abdulla, "Near real-time global solar radiation forecasting at multiple time-step horizons using the long short-term memory network," *Energies*, vol. 13, no. 14, p. 3517, Jul. 2020, doi: 10.3390/en13143517.

[23] H. Wang, Y. Liu, B. Zhou, C. Li, G. Cao, N. Voropai, and E. Barakhtenko, "Taxonomy research of artificial intelligence for deterministic solar power forecasting," *Energy Convers. Manage.*, vol. 214, Jun. 2020, Art. no. 112909, doi: 10.1016/j.enconman.2020.112909.

[24] Y. Feng, W. Hao, H. Li, N. Cui, D. Gong, and L. Gao, "Machine learning models to quantify and map daily global solar radiation and photovoltaic power," *Renew. Sustain. Energy Rev.*, vol. 118, Feb. 2020, Art. no. 109393, doi: 10.1016/j.rser.2019.109393.

[25] S. Mirjalili, "SCA: A sine cosine algorithm for solving optimization problems," *Knowl.-Based Syst.*, vol. 96, pp. 120–133, Mar. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950705115005043

[26] C. Wu, M. Khishe, M. Mohammadi, S. H. T. Karim, and T. A. Rashid, "Evolving deep convolutional neutral network by hybrid sine–cosine and extreme learning machine for real-time COVID19 diagnosis from X-ray images," *Soft Comput.*, vol. 2021, pp. 1–20, May 2021, doi: 10.1007/s00500-021-05839-6.

[27] Y. Wang, L. Yuan, M. Khishe, A. Moridi, and F. Mohammadzade, "Training RBF NN using sine-cosine algorithm for sonar target classification," *Arch. Acoust.*, vol. 45, no. 4, pp. 753–764, 2020. [Online]. Available: http://journals.pan.pl/Content/118266/PDF/aoa.2020.135281.pdf

[28] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: A gravitational search algorithm," *J. Inf. Sci.*, vol. 179, no. 13, pp. 2232–2248, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020025509001200

[29] M. R. Mosavi, M. Khishe, M. J. Naseri, G. R. Parvizi, and M. Ayat, "Multi-layer perceptron neural network utilizing adaptive best-mass gravitational search algorithm to classify sonar dataset," *Arch. Acoust.*, vol. 44, no. 1, pp. 137–151, 2019. [Online]. Available: http://journals.pan.pl/Content/110268/PDF/aoa.2019.126360.pdf

[30] D. H. Wolper and W. G. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, Apr. 1997, doi: 10.1109/4235.585893.

[31] M. M. Fouad, A. I. El-Desouky, R. Al-Hajj, and E.-S. M. El-Kenawy, "Dynamic group-based cooperative optimization algorithm," *IEEE Access*, vol. 8, pp. 148378–148403, 2020, doi: 10.1109/access.2020.3015892.

[32] A. Ibrahim, H. A. Ali, M. M. Eid, and E.-S.-M. El-kenawy, "Chaotic Harris hawks optimization for unconstrained function optimization," in *Proc. 16th Int. Comput. Eng. Conf. (ICENCO)*, Dec. 2020, pp. 153–158, doi: 10.1109/icenco49778.2020.9357403.

[33] E.-S. M. El-kenawy, S. Mirjalili, A. Ibrahim, M. Alrahmawy, M. El-Said, R. M. Zaki, and M. M. Eid, "Advanced meta-heuristics, convolutional neural networks, and feature selectors for efficient COVID-19 X-ray chest image classification," *IEEE Access*, vol. 9, pp. 36019–36037, 2021, doi: 10.1109/access.2021.3061058.

[34] J. Fan, L. Wu, X. Ma, H. Zhou, and F. Zhang, "Hybrid support vector machines with heuristic algorithms for prediction of daily diffuse solar radiation in air-polluted regions," *Renew. Energy*, vol. 145, pp. 2034–2045, Jan. 2020, doi: 10.1016/j.renene.2019.07.104.

[35] M. Noshy, A. Ibrahim, and H. A. Ali, "Optimization of live virtual machine migration in cloud computing: A survey and future directions," *J. Netw. Comput. Appl.*, vol. 110, pp. 1–10, May 2018, doi: 10.1016/j.jnca.2018.03.002.

[36] H. Zang, L. Liu, L. Sun, L. Cheng, Z. Wei, and G. Sun, "Short-term global horizontal irradiance forecasting based on a hybrid CNN-LSTM model with spatiotemporal correlations," *Renew. Energy*, vol. 160, pp. 26–41, Nov. 2020, doi: 10.1016/j.renene.2020.05.150.

[37] K. Nam, S. Hwangbo, and C. Yoo, "A deep learning-based forecasting model for renewable energy scenarios to guide sustainable energy policy: A case study of Korea," *Renew. Sustain. Energy Rev.*, vol. 122, Apr. 2020, Art. no. 109725, doi: 10.1016/j.rser.2020.109725.

[38] A. Ibrahim, S. Mohammed, H. A. Ali, and S. E. Hussein, "Breast cancer segmentation from thermal images based on chaotic salp swarm algorithm," *IEEE Access*, vol. 8, pp. 122121–122134, 2020, doi: 10.1109/access.2020.3007336.

[39] A. Ibrahim, A. Tharwat, T. Gaber, and A. E. Hassanien, "Optimized superpixel and AdaBoost classifier for human thermal face recognition," *Signal, Image Video Process.*, vol. 12, no. 4, pp. 711–719, May 2018, doi: 10.1007/s11760-017-1212-6.

[40] A. Mahmoodzadeh, M. Mohammadi, K. M G Noori, M. Khishe, H. H. Ibrahim, H. F. H. Ali, and S. N. Abdulhamid, "Presenting the best prediction model of water inflow into drill and blast tunnels among several machine learning techniques," *Autom. Construct.*, vol. 127, Jul. 2021, Art. no. 103719, doi: 10.1016/j.autcon.2021.103719.

[41] M. R. Mosavi, M. Khishe, Y. H. Khani, and M. Shabani, "Training radial basis function neural network using stochastic fractal search algorithm to classify sonar dataset," *Iranian J. Electr. Electron. Eng.*, vol. 13, no. 1, pp. 100–111, Mar. 2017, doi: 10.22068/IJEEE.13.1.10.

[42] R. Bello, Y. Gomez, A. Nowe, and M. M. Garcia, "Two-step particle swarm optimization to solve the feature selection problem," in *Proc. 7th Int. Conf. Intell. Syst. Des. Appl. (ISDA)*, Oct. 2007, pp. 691–696.

[43] A. Ibrahim, M. Noshy, H. A. Ali, and M. Badawy, "PAPSO: A power-aware VM placement technique based on particle swarm optimization," *IEEE Access*, vol. 8, pp. 81747–81764, 2020, doi: 10.1109/access.2020.2990828.

[44] E. M. Hassib, A. I. El-Desouky, L. M. Labib, and E.-S.-M. El-kenawy, "WOA + BRNN: An imbalanced big data classification framework using whale optimization and deep neural network," *Soft Comput.*, vol. 24, no. 8, pp. 5573–5592, Mar. 2019, doi: 10.1007/s00500-019-03901-y.

[45] S. S. M. Ghoneim, T. A. Farrag, A. A. Rashed, E.-S. M. El-Kenawy, and A. Ibrahim, "Adaptive dynamic meta-heuristics for feature selection and classification in diagnostic accuracy of transformer faults," *IEEE Access*, vol. 9, pp. 78324–78340, 2021, doi: 10.1109/access.2021.3083593.

[46] M. M. Kabir, M. Shahjahan, and K. Murase, "A new local search based hybrid genetic algorithm for feature selection," *Neurocomputing*, vol. 74, no. 17, pp. 2914–2928, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231211002748

[47] E.-S. M. El-Kenawy and M. Eid, "Hybrid gray wolf and particle swarm optimization for feature selection," *Int. J. Innov. Comput. Inf. Control*, vol. 16, no. 3, pp. 831–844, 2020.

[48] M. Jain, V. Singh, and A. Rani, "A novel nature-inspired algorithm for optimization: Squirrel search algorithm," *Swarm Evol. Comput.*, vol. 44, pp. 148–175, Feb. 2019, doi: 10.1016/j.swevo.2018.02.013.

[49] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019. [Online]. Available: https://doi.org/10.1016/j.future.2019.02.028

[50] Q. Li, H. Ning, J. Gong, X. Li, and B. Dai, "A hybrid greedy sine cosine algorithm with differential evolution for global optimization and cylindricity error evaluation," *Appl. Artif. Intell.*, vol. 35, no. 2, pp. 171–191, Dec. 2020, doi: 10.1080/08839514.2020.1848276.

[51] S. J. Rosli, H. A. Rahim, K. N. A. Rani, R. Ngadiran, R. B. Ahmad, N. Z. Yahaya, M. Abdulmalek, M. Jusoh, M. N. M. Yasin, T. Sabapathy, and A. M. Andrew, "A hybrid modified method of the sine cosine algorithm using Latin hypercube sampling with the cuckoo search algorithm for optimization problems," *Electronics*, vol. 9, no. 11, p. 1786, Oct. 2020, doi: 10.3390/electronics9111786.

[52] A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, "Marine predators algorithm: A nature-inspired Metaheuristic," *Expert Syst. Appl.*, vol. 152, Aug. 2020, Art. no. 113377, doi: 10.1016/j.eswa.2020.113377.

[53] M. Khishe and M. R. Mosavi, "Chimp optimization algorithm," *Expert Syst. Appl.*, vol. 149, Jul. 2020, Art. no. 113338, doi: 10.1016/j.eswa.2020.113338.

[54] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization," *Future Gener. Comput. Syst.*, vol. 111, pp. 300–323, Oct. 2020, doi: 10.1016/j.future.2020.03.055.

[55] *Hawai'i Space Exploration Analog and Simulation*. Accessed: Oct. 4, 2020. [Online]. Available: https://hi-seas.org

[56] *Solar Radiation Prediction Task From NASA Hackathon*. Accessed: Oct. 4, 2020. [Online]. Available: https://www.kaggle.com/dronio/SolarEnergy

[57] M. M. Eid, E.-S.-M. El-kenawy, and A. Ibrahim, "A binary sine cosine-modified whale optimization algorithm for feature selection," in *Proc. Nat. Comput. Colleges Conf. (NCCC)*, Mar. 2021, pp. 1–6, doi: 10.1109/nccc49330.2021.9428794.

[58] E. M. Hassib, A. I. El-Desouky, E.-S.-M. El-Kenawy, and S. M. El-Ghamrawy, "An imbalanced big data mining framework for improving optimization algorithms performance," *IEEE Access*, vol. 7, pp. 170774–170795, 2019.

[59] A. A. Nasser, M. Z. Rashad, and S. E. Hussein, "A two-layer water demand prediction system in urban areas based on micro-services and LSTM neural networks," *IEEE Access*, vol. 8, pp. 147647–147661, 2020, doi: 10.1109/access.2020.3015655.

[60] E.-S. M. El-Kenawy, M. M. Eid, M. Saber, and A. Ibrahim, "MbGWO-SFS: Modified binary grey wolf optimizer based on stochastic fractal search for feature selection," *IEEE Access*, vol. 8, no. 1, pp. 107635–107649, 2020, doi: 10.1109/access.2020.3001151.

[61] E.-S. M. El-kenawy, A. Ibrahim, S. Mirjalili, M. M. Eid, and S. E. Hussein, "Novel feature selection and voting classifier algorithms for COVID-19 classification in CT images," *IEEE Access*, vol. 8, pp. 179317–179335, 2020, doi: 10.1109/access.2020.3028012.

[62] B. Kamiński, M. Jakubczyk, and P. Szufel, "A framework for sensitivity analysis of decision trees," *Central Eur. J. Oper. Res.*, vol. 26, no. 1, pp. 135–159, May 2017, doi: 10.1007/s10100-017-0479-6.

[63] R. Al-Hajj, A. Assi, and M. M. Fouad, "Stacking-based ensemble of support vector regressors for one-day ahead solar irradiance prediction," in *Proc. 8th Int. Conf. Renew. Energy Res. Appl. (ICRERA)*, Nov. 2019, pp. 428–433, doi: 10.1109/icrera47325.2019.8996629.

EL-SAYED M. EL-KENAWY (Member, IEEE) is currently an Assistant Professor with the Delta Higher Institute for Engineering and Technology (DHIET), Mansoura, Egypt. He is inspiring and motivating students by providing a thorough understanding of a variety of computer concepts. He has published over 30 publications with over 700 citations and an H-index of 19. He has pioneered and launched independent research programs. He adept at sometimes explaining complex concepts in an easy-to-understand manner. His research interests include optimization, artificial intelligence, machine learning, deep learning, data science, and digital marketing. He serves as a Reviewer for the journal IEEE ACCESS.

SEYEDALI MIRJALILI (Senior Member, IEEE) is currently the Director of the Centre for Artificial Intelligence Research and Optimization, Torrens University Australia at Brisbane. He is internationally recognized for his advances in swarm intelligence and optimization, including the first set of algorithms from a synthetic intelligence standpoint—a radical departure from how natural systems are typically understood and a systematic design framework to reliably benchmark, evaluate, and propose computationally cheap robust optimization algorithms. He has published over 200 publications with over 20,000 citations and an H-index of 51. As the most cited researcher in robust optimization, he is in the list of 1% highly cited researchers and named as one of the most influential researchers in the world by the Web of Science. He is working on the applications of multi-objective and robust meta-heuristic optimization techniques as well. His research interests include robust optimization, engineering optimization, multi-objective optimization, swarm intelligence, evolutionary algorithms, and artificial neural networks. He is an Associate Editor of several journals, including *Neurocomputing*, *Applied Soft Computing*, *Advances in Engineering Software*, *Applied Intelligence*, *PLOS One*, and IEEE ACCESS.

SHERIF S. M. GHONEIM (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees from the Faculty of Engineering at Shoubra, Zagazig University, Egypt, in 1994 and 2000, respectively, and the Ph.D. degree in electrical power and machines from the Faculty of Engineering, Cairo University, in 2008. Since 1996, he has been teaching at the Faculty of Industrial Education, Suez Canal University, Egypt. From 2005 to 2007, he was a Guest Researcher with the Institute of Energy Transport and Storage (ETS), University of Duisburg–Essen, Germany. He joined Taif University as an Associate Professor with the Department of Electrical Engineering, Faculty of Engineering. His research interests include grounding systems, dissolved gas analysis, breakdown in SF6 gas, and AI technique applications.

MARWA METWALLY EID received the Ph.D. degree in electronics and communications engineering from the Faculty of Engineering, Mansoura University, Egypt, in 2015. She has been working as an Assistant Professor with the Delta Higher Institute for Engineering and Technology, since 2011. Her current research interests include image processing, encryption, wireless communication systems, and field programmable gate array (FPGA) applications.

M. EL-SAID received the B.Sc. (Hons.), M.Sc., and Ph.D. degrees from Mansoura University, Egypt, in 1981, 1987, and 1992, respectively, all in electric power engineering. Since 2005, he has been a Professor of electric power systems with the Department of Electrical Engineering, Faculty of engineering, Mansoura University. He had held several administrative positions. He was the Vice Dean for community services and environmental development at the Faculty of Engineering, Mansoura University, Egypt, from 2010 to 2011. He was also the Director of the Communication and Information Technology Centre (CITC), Mansoura University, from 2014 to 2016. From 2016 to 2018, he was the Dean of the Faculty of engineering, Mansoura University. He is currently the Dean of the Delta Higher Institute of Engineering and Technology, Ministry of Higher Education, Egypt. His main research interests include power system operation and control, smart grids, and renewable energy. He is an Official Reviewer of Egyptian Universities Promotion Committees, Supreme Council of Universities, Egypt. He is a Regular Reviewer of many scientific journals, such as IEEE TRANSACTIONS ON POWER SYSTEMS, IEEE TRANSACTIONS ON POWER DELIVERY, and *Electric Power Components and Systems*.

ZEESHAN SHAFI KHAN is currently working as an Assistant Professor with the College of Engineering and Computer Science, Mustaqbal University, Saudi Arabia. His research interests include network communication and cyber-security, multimedia services, and machine learning. He has published a number of articles in well-reputed journals and high-ranked conferences. He is a member of steering committees of various journals and served as a guest editor for few well-known journals.

ABDELHAMEED IBRAHIM (Member, IEEE) received the bachelor's and master's degrees in engineering from the Department of Computer Engineering and Systems, in 2001 and 2005, respectively, and the Ph.D. degree in engineering from the Faculty of Engineering, Chiba University, Japan, in 2011. He was with the Faculty of Engineering, Mansoura University, Egypt, from 2001 to 2007, where he is currently an Associate Professor of computer engineering with the Faculty of Engineering. He has published over 50 publications with over 1000 citations and an H-index of 19. His research interests include pattern recognition, optimization, machine learning, virtualization, and live virtual machine migration. He serves as a Reviewer for the *Journal of Electronic Imaging*, *Optical Engineering*, the IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS, IEEE ACCESS, *Computer Standards and Interfaces*, *Journal of Healthcare Engineering*, *IET Image Processing*, *Multimedia Tools and Applications*, and other respected journals.

● ● ●