Journal of Artificial Intelligence and Metaheuristic

♦ ISSN: 2833-5597 Vol: 11, No. 01, 2026 ♦ PP. 56–67
Article DOI: https://doi.org/10.54216/JAIM.110106

FULL LENGTH ARTICLE

An Enhanced LSTM Framework for Solar Radiation Forecasting Using Hybrid Grey Wolf–Waterwheel Plant Optimization

Alaa Mohamed Abdel-Moati[1,*] **Abdullah Muhammad Ibrahim**[2]

[1] Department of Architecture Engineering, Delta Higher Institute for Engineering & Technology (DHIET), Mansoura, Egypt
[2] Department of Civil Engineering, Delta Higher Institute for Engineering & Technology (DHIET), Mansoura, Egypt

Emails: CH2200004@dhiet.edu.eg · CH2400169@dhiet.edu.eg

Received: August 22, 2025 Revised: October 18, 2025 Accepted: December 20, 2025 ⋆ Corresponding author

ABSTRACT

Accurate solar radiation prediction is critical for optimizing renewable energy utilization, yet traditional machine-learning models struggle with suboptimal forecasting accuracy. To address this challenge, we propose a novel hybrid optimization algorithm, the Grey Wolf and Water Whale Plant Optimizer (GWWWPA), to enhance Long Short-Term Memory (LSTM) networks for improved solar radiation forecasting. The proposed method leverages the exploration-exploitation synergy of the Grey Wolf Optimizer (GWO) and Water Whale Plant Algorithm (WWPA) to optimize LSTM hyperparameters effectively. Experimental results on the NASA Space Apps Moscow dataset demonstrate that GWWWPA-LSTM outperforms existing optimization techniques, achieving the lowest Mean Squared Error (MSE) of 0.00019392 and the highest R^2 score of 0.917262, surpassing standalone GWO, WWPA, PSO, and WOA. These findings highlight the potential of hybrid metaheuristic approaches in enhancing predictive accuracy, facilitating more reliable solar energy forecasting, and supporting sustainable energy management systems.

Keywords: Hybrid Optimization ▪ Grey Wolf Optimizer ▪ Water Whale Plant Algorithm ▪ Solar Radiation Prediction ▪ Long Short-Term Memory (LSTM)

1. INTRODUCTION

The global transition toward renewable energy sources has become a critical priority due to the depletion of fossil fuels and the urgent need to address climate change. Solar energy, in particular, has garnered significant attention due to its abundance, sustainability, and potential to reduce dependence on non-renewable energy sources. Solar photovoltaic (PV) systems have been widely adopted for electricity generation; however, their efficiency is highly dependent on the availability of solar radiation, which varies due to atmospheric and meteorological conditions [1, 2]. These fluctuations pose significant challenges for power grid operators, energy planners, and businesses that rely on solar power. Accurate solar radiation forecasting is essential for optimizing energy production, enhancing grid integration, reducing energy costs, and ensuring the reliability of solar power plants [3, 4].

Despite the increasing availability of meteorological datasets and advancements in predictive modeling techniques, accurately forecasting solar radiation remains a complex challenge. Various environmental factors, such as temperature, humidity, wind speed, barometric pressure, and cloud cover, influence the intensity and distribution of solar radiation [5]. The non-linearity and randomness of these variables make traditional statistical models insufficient for capturing intricate dependencies [6]. As a result, machine learning and deep learning approaches have emerged as powerful alternatives due to their ability to learn complex patterns from historical data [7].

Among machine learning techniques, deep learning models, particularly Long Short-Term Memory (LSTM) networks, have shown remarkable potential in time-series forecasting applications, including solar radiation prediction [8]. LSTM networks are well-suited for capturing long-term dependencies and addressing the vanishing gradient problem, which is prevalent in traditional recurrent neural networks (RNNs) [9, 10]. However, their performance heavily relies on selecting optimal hyperparameters, such as the number of hidden layers, learning rate, batch size, and activation functions. Suboptimal hyperparameters can lead to slow convergence, overfitting, or underfitting, thereby reducing the model's predictive accuracy [11, 12].

To address this challenge, optimization algorithms have been widely adopted for fine-tuning machine learning models. Inspired by natural and evolutionary processes, meta-heuristic optimization techniques have demonstrated superior capabilities in navigating complex search spaces to identify optimal solutions [13]. These algorithms use stochastic approaches to balance exploration and exploitation, making them particularly useful for hyperparameter optimization in deep learning models.

The Grey Wolf Optimizer (GWO) is a nature-inspired metaheuristic algorithm that models the leadership hierarchy and cooperative hunting strategies of grey wolves [14]. GWO has demonstrated strong performance in various optimization problems due to its effective global search capabilities and convergence efficiency. However, despite its strengths, GWO is sometimes prone to premature convergence and may become trapped in local optima, particularly when navigating highly nonlinear search spaces.

The Water Whale Plant Algorithm (WWPA) is another bio-inspired optimization technique derived from the cooperative foraging behavior of whales in freshwater vegetation systems [15]. This algorithm enhances search diversification by integrating adaptive foraging mechanisms, ensuring a balanced trade-off between exploration and exploitation. Although WWPA has shown promising results in optimization tasks, it may exhibit instability when dealing with dynamic search environments.

Given these algorithms' strengths and weaknesses, hybrid optimization strategies have emerged as promising solutions to enhance performance. Hybrid metaheuristic approaches combine multiple algorithms to leverage their complementary advantages, leading to improved convergence speed, solution quality, and robustness. This study introduces a novel hybrid optimization framework: the Grey Wolf and Water Whale Plant Optimizer (GWWWPA), which integrates GWO and WWPA to optimize solar radiation forecasting models.

This research aims to develop an optimization framework that enhances LSTM-based solar radiation prediction by addressing the limitations of standalone optimization techniques. By incorporating the leadership-driven exploration of GWO and the adaptive foraging behavior of WWPA, the hybrid GWWWPA approach aims to refine hyperparameter selection, thereby improving predictive accuracy and computational efficiency. Given the increasing reliance on solar energy systems, an accurate forecasting model can significantly impact the energy sector by facilitating better resource allocation, minimizing energy losses, and improving solar power integration into smart grids.

This study utilizes real-world meteorological data from the NASA Space Apps Moscow dataset. The dataset includes critical atmospheric variables such as wind direction, wind speed, humidity, and temperature, making it an ideal benchmark for evaluating the effectiveness of the proposed optimization framework. This research contributes to the ongoing efforts to enhance solar energy forecasting by developing an advanced predictive model that integrates hybrid optimization techniques. It supports the global transition toward sustainable energy solutions.

This study presents a novel hybrid optimization algorithm, the Grey Wolf and Water Whale Plant Optimizer (GWWWPA), designed to enhance solar radiation forecasting through optimized hyperparameter selection in Long Short-Term Memory (LSTM) networks. The primary contributions of this research are as follows:

- **Hybridization of GWO and WWPA:** We integrate the Grey Wolf Optimizer (GWO) and the Water Whale Plant Algorithm (WWPA) to balance exploration and exploitation effectively, mitigating their limitations and improving global search capabilities.
- **Optimization of LSTM for Solar Radiation Forecasting:** The proposed GWWWPA method optimally tunes the hyperparameters of LSTM networks, enhancing predictive accuracy and stability in complex meteorological datasets.
- **Performance Superiority:** The experimental evaluation, conducted on the NASA Space Apps Moscow dataset, demonstrates that GWWWPA-LSTM outperforms existing optimization techniques, achieving superior results in terms of Mean Squared Error (MSE) and R-squared (R^2) metrics.
- **Computational Efficiency:** Compared to standalone GWO, WWPA, and other metaheuristic approaches, the hybrid GWWWPA achieves faster convergence and improved robustness in high-dimensional search spaces.
- **Practical Implications:** The enhanced forecasting capability of GWWWPA-LSTM can significantly benefit solar energy management, smart grid integration, and sustainable energy planning by providing more reliable predictions of solar radiation levels.

Thus, the proposed hybrid optimization approach significantly advances solar energy forecasting. It demonstrates the potential of metaheuristic hybridization to improve deep learning model performance.

The remainder of this paper is structured as follows: Section 2 presents a comprehensive literature review of optimization techniques for solar radiation forecasting. Section 3 describes the dataset and preprocessing steps to ensure data quality and consistency. Section 4 introduces the proposed GWWWPA algorithm, detailing its mathematical formulation and hybridization strategy. Section 5 discusses the experimental setup, including model configurations and performance evaluation metrics. Section 6 presents the comparative analysis results, demonstrating the proposed approach's effectiveness. Finally, Section 7 concludes the paper with key findings and potential directions for future research.

2. LITERATURE REVIEW

The increasing demand for sustainable energy sources has led to significant research into predicting solar radiation (SR). Accurate solar irradiance forecasting is essential for optimizing photovoltaic (PV) power generation and ensuring grid stability. While various machine learning (ML) and deep learning (DL) models have been explored, integrating metaheuristic optimization algorithms has increasingly been investigated to enhance predictive accuracy.

Several studies have demonstrated the effectiveness of ML and DL models in predicting SR. Traditional techniques widely adopted in the literature include Support Vector Machines (SVM) [16], Convolutional Neural Networks (CNN) [17], Multilayer Perceptron (MLP) [18], and Long Short-Term Memory (LSTM) networks. Furthermore, deep structured architectures incorporating hybrid models, such as Deep Neural Networks (DNN) and Recurrent Neural Networks (RNN), have been proposed for solar irradiance forecasting [18].

Feature extraction plays a crucial role in improving prediction accuracy. Several statistical and dimensionality reduction methods have been employed to refine input data before training ML models [18], including Independent Component Analysis (ICA), Principal Component Analysis (PCA), and Linear Discriminant Analysis (LDA). Probabilistic forecasting methods have also been explored, providing confidence intervals for predictions to support decision-making in solar energy systems [19].

Metaheuristic optimization techniques have been increasingly incorporated into ML models to optimize hyperparameters and enhance predictive performance. Studies have highlighted the use of Particle Swarm Optimization (PSO), Genetic Algorithm (GA), and Grey Wolf Optimization (GWO) for tuning neural network architectures and improving prediction accuracy [16, 17, 20]. A hybrid approach combining GWO and the Deer Hunting Optimization Algorithm (DHOA), termed Grey Updated DHOA (GU-DHOA), was proposed to optimize the number of hidden neurons in neural networks, significantly improving solar irradiance prediction [17].

Other studies have explored the Reptile Search Algorithm (RSA) for tuning hyperparameters in LSTM and bidirectional LSTM (BiLSTM) models, achieving superior results compared to traditional ML models [21]. Hybrid models combining Ant Colony Optimization (ACO), Cuckoo Search (CS), and GWO with SVM have demonstrated higher accuracy in solar radiation prediction across various climatic regions [20].

Adaptive Neuro-Fuzzy Inference Systems (ANFIS) hybridized with Salp Swarm Algorithm (SSA) and Grasshopper Optimization Algorithm (GOA) have also been introduced, achieving notable improvements over classical ANFIS models for global SR prediction [22]. Moreover, the Electromagnetic Field Optimization (EFO) algorithm has been employed to fine-tune neural networks, surpassing benchmark optimization techniques such as Shuffled Complex Evolution and Shuffled Frog Leaping Algorithm [23].

Despite the progress in SR prediction, several challenges remain. The variability of solar irradiance due to meteorological factors necessitates robust predictive models capable of generalizing across different geographic regions and time horizons [24, 25]. A systematic review highlighted that while ML-based forecasters show promising results, their accuracy can be further enhanced by integrating metaheuristics for hyperparameter tuning and feature selection [25].

Optimization of solar energy systems is also crucial for reducing operational costs and enhancing energy efficiency. Hybrid optimization techniques have demonstrated the potential to improve system performance by optimizing solar absorption cooling systems and reducing power losses [24]. Future research should focus on developing more efficient hybrid metaheuristic algorithms that balance exploration and exploitation while minimizing computational overhead.

Integrating ML and DL models with metaheuristic optimization algorithms has significantly improved the accuracy of SR prediction. Techniques such as PSO, GWO, and hybrid optimization approaches have demonstrated substantial performance gains. However, challenges remain in ensuring model generalization across different climatic conditions. Future research should focus on developing adaptive hybrid models that leverage multiple optimization techniques to enhance predictive performance further and support the global transition to sustainable energy solutions.

3. DATASET AND PREPROCESSING

3.1 Dataset Description

The dataset used in this study originates from the NASA Space Apps Moscow 2016 Challenge, an international hackathon aimed at utilizing open space and Earth science data to address real-world challenges [26]. The dataset consists of meteorological observations recorded at the HI-SEAS weather station over four months, from September to December 2016, during the interval between Mission IV and Mission V. The primary objective of this dataset is to facilitate the prediction of solar radiation levels based on atmospheric conditions, which is essential for optimizing solar energy systems.

The dataset includes multiple meteorological parameters that influence solar radiation. These parameters capture essential environmental factors affecting the intensity and distribution of sunlight. The key features and their descriptions are as follows:

- **UNIXTime** – A timestamp representing the number of seconds since January 1, 1970, ensuring the records' chronological ordering.
- **Date (yyyy-mm-dd)** – The date on which the meteorological measurements were recorded.
- **Time (hh:mm:ss)** – The local time of measurement in a 24-hour format.
- **Solar Radiation (W/m²)** – The target variable representing the intensity of solar energy received per unit area.
- **Temperature (°F)** – The ambient temperature of the atmosphere, recorded in degrees Fahrenheit.
- **Humidity (%)** – The relative humidity of the air, expressed as a percentage.

- **Barometric Pressure (Hg)** – The atmospheric pressure measured in inches of mercury.
- **Wind Direction (Degrees)** – The direction from which the wind is blowing, measured in degrees from true north (0° to 360°).
- **Wind Speed (mph)** – The speed of the wind, measured in miles per hour.
- **Sunrise Time** – The recorded local time of sunrise in Hawaii.
- **Sunset Time** – The recorded local sunset time in Hawaii.

This dataset provides a comprehensive set of meteorological features, making it highly suitable for developing predictive models to estimate solar radiation levels.

3.2 Data Cleaning and Transformation

Preprocessing the dataset is crucial before training machine learning models to ensure data quality, consistency, and reliability. Several preprocessing steps are applied, including handling missing values, detecting and removing anomalies, normalizing the data, and performing feature engineering.

3.2.1 Handling Missing Values

Missing data is common in real-world meteorological datasets, often resulting from sensor malfunctions, transmission errors, or unfavorable weather conditions. In this dataset, missing values may appear in any numerical attribute, particularly temperature, humidity, wind speed, and solar radiation. Several strategies are applied to address this issue:

- **Linear Interpolation:** Missing values in continuous variables (temperature, humidity, wind speed) are estimated using linear interpolation, which assumes that a linear trend between known values can approximate missing data points.
- **Forward-Filling:** If a missing value occurs in a time-dependent variable, it is replaced with the most recent valid observation.
- **Removal of Highly Incomplete Records:** Records with more than 30% missing data are removed to maintain the integrity of the dataset.

3.2.2 Detection and Removal of Anomalies

Meteorological data may contain anomalies due to sensor errors or extreme weather conditions. The following techniques are used to identify and handle anomalies:

- **Interquartile Range (IQR) Method:** Outliers are identified as values exceeding 1.5 times the interquartile range from the first and third quartiles.
- **Z-Score Analysis:** Observations with a Z-score greater than 3 (indicating significant deviation from the mean) are flagged as potential anomalies.
- **Domain Knowledge-Based Filtering:** Unphysiological values, such as negative wind speed or temperatures exceeding reasonable limits, are removed.

3.2.3 Feature Scaling and Normalization

Machine learning models, particularly those involving deep learning, benefit from feature normalization to ensure faster convergence and improved generalization. The following transformations are applied:

- **Min-Max Scaling:** Features are scaled to a range of $[0,1]$ using:
$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{1}$$
- **Log Transformation:** To stabilize variance, a log transformation is applied to highly skewed variables such as solar radiation.

3.2.4 Feature Engineering

Feature engineering is performed to enhance the predictive capability of the dataset:

- **Interaction Features:** Temperature-Humidity interactions are introduced to capture nonlinear dependencies.
- **Time-based Features:** The UNIX timestamp is decomposed into hour-of-day and day-of-year components to account for diurnal and seasonal variations.
- **Wind Components:** Wind direction is decomposed into U and V components for better representation in regression models.

Figure 1 showcases the time-series plot of solar radiation data. This figure visualizes solar radiation measurements' temporal patterns and variability over the four-month dataset, understanding seasonal and daily trends.

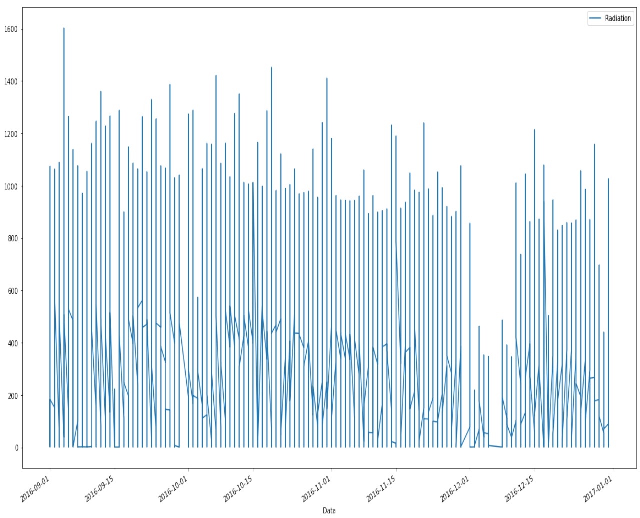

Figure 1. Time-series plot of solar radiation data.

Figure 2 provides a boxplot of meteorological features grouped by temperature. It demonstrates the distributions of radiation, pressure, humidity, wind direction, and speed across different temperature ranges, providing insights into each feature's variability and potential outliers.

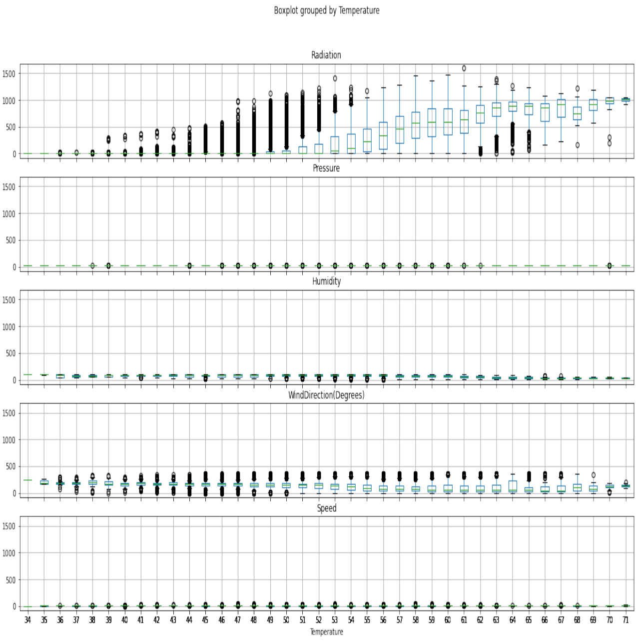

Figure 2. Boxplot of meteorological features grouped by temperature.

Figure 3 illustrates the correlation heatmap for the meteorological dataset. This heatmap highlights the relationships between radiation, temperature, pressure, humidity, wind direction, and speed. Positive and negative correlations are represented by shades of green and red, respectively, aiding in feature selection and dependency analysis.

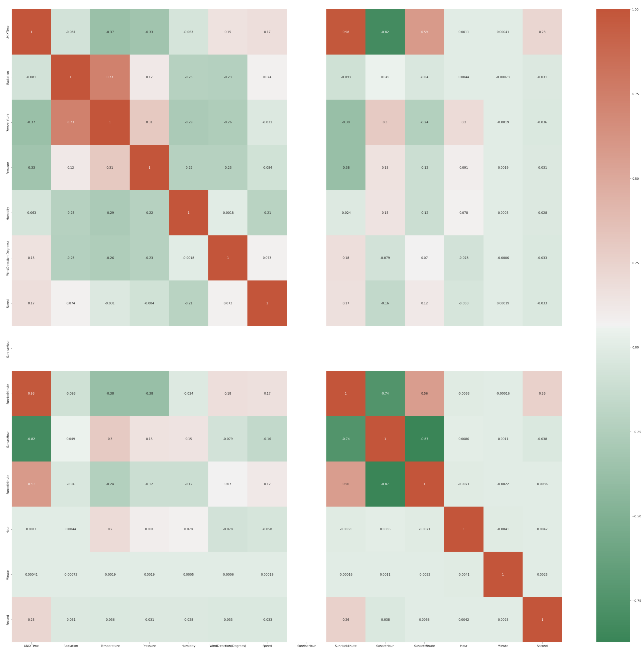

Figure 3. Correlation heatmap for the meteorological dataset.

This section presents a comprehensive dataset overview, including its attributes, preprocessing steps, and exploratory analysis. The cleaned and processed dataset is now ready for use in machine learning models, and optimized features ensure improved model performance. The following section describes the proposed hybrid optimization approach for enhancing solar radiation forecasting.

4. PROPOSED HYBRID OPTIMIZATION ALGORITHM: GWWWPA

4.1 Grey Wolf Optimizer (GWO)

4.1.1 Mathematical Formulation

The Grey Wolf Optimizer (GWO) is a metaheuristic optimization algorithm inspired by the social hierarchy and hunting mechanism of grey wolves *(Canis lupus)* [27]. This algorithm models the leadership dominance within a wolf pack and their cooperative behavior in tracking, encircling, and attacking prey. In GWO, the population of candidate solutions (wolves) is structured into four levels: alpha (α), beta (β), delta (δ), and omega (ω), where α represents the best candidate solution, followed by β and δ which assist in hunting, while ω wolves follow the lead of the dominant ones.

Mathematically, the position update mechanism of GWO is modeled using the following equations:

$$\vec{D} = \left|\vec{C} \cdot \vec{X}_p(t) - \vec{X}(t)\right| \tag{2}$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \tag{3}$$

Where:

- $\vec{X}(t)$ represents the position vector of a grey wolf at iteration t.
- $\vec{X}_p(t)$ denotes the position vector of the prey (i.e., the best solution found so far).
- $\vec{A}$ and $\vec{C}$ are coefficient vectors that control exploration and exploitation.
- $\vec{D}$ denotes the distance vector between the prey and the current wolf.

The vectors $\vec{A}$ and $\vec{C}$ are computed as:

$$\vec{A} = 2 \cdot \vec{a} \cdot \vec{r}_1 - \vec{a} \tag{4}$$

$$\vec{C} = 2 \cdot \vec{r}_2 \tag{5}$$

Where $\vec{a}$ is a control parameter that linearly decreases from 2 to 0 over iterations, while $\vec{r}_1$ and $\vec{r}_2$ are random vectors in the range $[0, 1]$.

4.1.2 Leadership Hierarchy and Hunting Mechanism

The hunting mechanism of GWO is based on the leadership dominance in a grey wolf pack. The top three solutions (α, β, and δ) guide the movement of the entire population by updating each wolf's position relative to the leaders. This is mathematically formulated as:

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 \cdot \vec{D}_\alpha \tag{6}$$

$$\vec{X}_2 = \vec{X}_\beta - \vec{A}_2 \cdot \vec{D}_\beta \tag{7}$$

$$\vec{X}_3 = \vec{X}_\delta - \vec{A}_3 \cdot \vec{D}_\delta \tag{8}$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \tag{9}$$

Where:

- $\vec{X}_\alpha$, $\vec{X}_\beta$, and $\vec{X}_\delta$ are the best three solutions.
- $\vec{A}_1$, $\vec{A}_2$, and $\vec{A}_3$ are randomly adjusted parameters.
- $\vec{D}_\alpha$, $\vec{D}_\beta$, and $\vec{D}_\delta$ are the distance vectors from α, β, and δ wolves, respectively.

This approach ensures that the search agents navigate towards promising regions in the solution space while maintaining a balance between exploration (diversification) and exploitation (intensification).

4.1.3 Encircling and Attacking the Prey

The GWO algorithm simulates the encircling behavior of wolves using the aforementioned position update equations. As iterations progress, the value of $\vec{a}$ decreases linearly, reducing the range of exploration and focusing the search on the most promising areas. The transition from exploration to exploitation is achieved through the following conditions:

- When $|\vec{A}| > 1$, wolves explore the search space randomly, increasing diversity.
- When $|\vec{A}| < 1$, wolves intensify their search around the prey (best solutions).

This mechanism allows GWO to efficiently locate global optima while minimizing the risk of premature convergence.

4.1.4 Advantages of GWO

The key advantages of GWO in solving optimization problems include:

- **Simple and Flexible:** GWO requires only a minimal set of tunable parameters, making it easy to implement and adapt for various optimization problems.
- **Exploration and Exploitation Balance:** The adaptive control of $\vec{a}$ enables a seamless transition from global search to local refinement.
- **Derivative-Free Mechanism:** Since GWO does not require derivative information, it performs effectively on non-differentiable and complex optimization tasks.
- **High Convergence Speed:** The cooperative search mechanism ensures rapid convergence while minimizing the likelihood of getting trapped in local optima.

Extensive research has demonstrated that GWO delivers excellent results for benchmark test functions and real-world engineering optimization problems.

This section has comprehensively discussed the Grey Wolf Optimizer, including its theoretical foundation, hunting strategy, and key advantages. In the next section, we introduce the proposed hybrid algorithm, *Grey Wolf and Water Whale Plant Optimizer (GWWWPA)*, which enhances GWO by incorporating additional exploration strategies derived from the *Water Whale Plant Algorithm (WWPA)*.

4.2 Water Whale Plant Algorithm (WWPA)

4.2.1 Inspiration from Whale Behavior in Water Plant Optimization

The Water Whale Plant Algorithm (WWPA) is a recently developed metaheuristic optimization technique inspired by the foraging behavior of waterwheel plants, scientifically known as *Aldrovanda vesiculosa* [28]. These carnivorous aquatic plants exhibit unique prey-trapping mechanisms, which are the foundation for designing an efficient optimization algorithm. The fundamental principle of WWPA is to model these plants' coordinated movements and prey-capturing strategies, leveraging their ability to maximize food capture while minimizing energy expenditure.

Waterwheel plants are characterized by their ability to float freely in water bodies while their specialized leaf traps snap shut when triggered by prey. This rapid snapping action and subsequent digestion process represent an optimal balance between exploration (searching for new prey) and exploitation (trapping and processing food). By incorporating these principles, WWPA emulates the dynamic adaptation and efficient decision-making observed in nature to enhance global and local search capabilities within an optimization framework.

4.2.2 Exploration vs. Exploitation Trade-Off

The effectiveness of any metaheuristic algorithm depends on its ability to balance exploration and exploitation. Exploration ensures that the algorithm effectively searches the entire solution space to avoid local optima, while exploitation refines the search around promising solutions to enhance convergence speed. WWPA achieves this balance through two primary phases:

- **Exploration (Position Identification and Hunting of Insects):** During this phase, the waterwheel plant dynamically adjusts its position to locate potential prey. This behavior is simulated in the optimization process by allowing candidate solutions to move across the search space stochastic. The movement is governed by adaptive step sizes and randomized directional shifts, ensuring diversity in the search.
- **Exploitation (Carrying the Insect in the Suitable Tube):** Once prey is captured, the plant processes it for nutrient absorption. In WWPA, this corresponds to refining promising solutions by reducing randomness and focusing on fine-tuning the best candidates. This step ensures convergence toward the global optimum while avoiding premature stagnation.

Mathematically, the WWPA algorithm updates candidate solutions (**search agents**) using the following equations:

$$\vec{X}(t+1) = \vec{X}(t) + r_1 \cdot \left(\vec{X}_{best} - \vec{X}(t)\right) + r_2 \cdot \left(\vec{X}_{rand} - \vec{X}(t)\right) \tag{10}$$

Where:

- $\vec{X}(t)$ represents the current position of a candidate solution at iteration t.
- $\vec{X}_{best}$ is the best solution found so far.

- $\vec{X}_{rand}$ is a randomly selected candidate solution.
- r_1 and r_2 are random coefficients that control exploration and exploitation.

Additionally, to prevent premature convergence and enhance search diversity, WWPA employs a dynamic control parameter, a, which linearly decreases over iterations:

$$a(t) = 2 - \frac{2t}{T} \tag{11}$$

Where T is the total number of iterations, this parameter ensures a smooth transition from exploration to exploitation.

4.2.3 Advantages of WWPA

WWPA offers several key advantages over traditional optimization techniques:

- **Adaptive Search Mechanism:** The algorithm dynamically adjusts search intensity based on the distribution of candidate solutions, ensuring efficient exploration.
- **Efficient Convergence:** The balance between searching new areas and refining existing solutions enables fast convergence without getting trapped in local optima.
- **Robust Performance:** WWPA demonstrates reliable results across various optimization problems, including unimodal and multimodal functions.
- **Scalability:** The algorithm effectively handles large-scale optimization tasks and is well-suited for real-world engineering applications.

WWPA models its optimization strategy based on the foraging behavior of waterwheel plants in their natural ecosystem. The algorithm enhances search efficiency and solution accuracy by balancing exploration and exploitation dynamically. The following section discusses how WWPA and the Grey Wolf Optimizer (GWO) are integrated to improve optimization performance further.

4.3 Hybridization Strategy: GWWWPA

4.3.1 Algorithm Design: Combining the Advantages of GWO and WWPA

The Grey Wolf and Water Whale Plant Optimizer (GWWWPA) is a novel hybrid optimization algorithm that integrates the Grey Wolf Optimizer (GWO) and the Water Whale Plant Algorithm (WWPA). The primary motivation for this hybridization is to leverage the strengths of both metaheuristic approaches while mitigating their limitations.

- **Grey Wolf Optimizer (GWO):** Maintains a strong balance between exploration and exploitation by mimicking grey wolves' hierarchical leadership and cooperative hunting behavior. However, it may exhibit premature convergence when navigating complex search spaces.
- **Water Whale Plant Algorithm (WWPA):** Inspired by the prey-trapping mechanisms of waterwheel plants, WWPA introduces dynamic adaptation and an efficient diversification strategy to prevent stagnation in suboptimal solutions.

By integrating these two approaches, GWWWPA enhances global search capabilities while refining fine-tuning mechanisms, ultimately improving optimization performance. This hybrid strategy is particularly effective in optimizing hyperparameters for Long Short-Term Memory (LSTM) networks in solar radiation forecasting applications.

4.3.2 Optimization Function for LSTM Hyperparameters

The proposed hybrid optimizer is applied to optimize the hyperparameters of LSTM networks, which play a critical role in solar radiation prediction. The following parameters are optimized:

- Number of LSTM layers (h): Controls the depth of the network.
- Number of hidden units (n_h): Determines the complexity of learned representations.
- Learning rate (η): Affects the convergence speed of the model.
- Batch size (b): Defines the number of training samples per iteration.
- Dropout rate (d): Regularization parameter to prevent overfitting.

The hybrid optimizer updates these hyperparameters iteratively using a fitness function based on Mean Squared Error (MSE):

$$\text{Fitness} = \frac{1}{N} \sum_{i=1}^{N} (Y_i - \hat{Y}_i)^2 \tag{12}$$

where Y_i is the actual solar radiation, $\hat{Y}_i$ is the predicted output from LSTM, and N represents the total number of data points.

4.3.3 Advantages of GWWWPA

The integration of GWO and WWPA in GWWWPA offers the following benefits:

- Improved Exploration: The WWPA component enhances global search capabilities, ensuring better diversity.
- Strong Exploitation: The GWO component focuses on refining solutions using the leadership-based hunting strategy.
- Faster Convergence: The dynamic parameter adaptation in WWPA prevents stagnation, leading to faster convergence to optimal solutions.
- Robust Performance: The hybrid model is more effective for tuning LSTM hyperparameters, resulting in better generalization for solar radiation prediction.

This section introduced the GWWWPA hybrid optimizer, which combines GWO's structured leadership hierarchy with WWPA's adaptive diversification strategy. The following section presents experimental results demonstrating that GWWWPA is superior to standalone optimization methods.

5. IMPLEMENTATION AND EXPERIMENTAL SETUP

5.1 Machine Learning Models Used for Benchmarking

To evaluate the effectiveness of the proposed hybrid optimization algorithm (GWWWPA), we compare the performance of various machine learning models in solar radiation forecasting. The following models are used as benchmarks:

- **Long Short-Term Memory (LSTM)**: A recurrent neural network (RNN) architecture capable of learning long-term dependencies in time-series data. LSTM is selected as the primary model due to its ability to capture temporal patterns in meteorological data.
- **Extreme Gradient Boosting (XGBoost)**: A robust gradient-boosting algorithm known for its speed and efficiency in handling structured data.
- **Categorical Boosting (CatBoost)**: A gradient-boosting model optimized for categorical data, reducing overfitting and improving generalization.
- **Multi-Layer Perceptron Regressor (MLPRegressor)**: A fully connected feedforward neural Multilayerd for function approximation and regression.
- **Support Vector Regression (SVR)**: A kernel-based regression model that minimizes prediction error while maximizing margin.
- **K-Nearest Neighbors (KNN)**: A non-parametric algorithm that predicts values based on similarity to nearest training data points.
- **Linear Regression (LR)**: A simple statistical model that assumes a linear relationship between input features and the target variable.

These models are evaluated with and without optimization to assess the proposed GWWWPA approach's impact on performance improvement.

5.2 Optimization Framework

The proposed optimization framework aims to fine-tune hyperparameters for LSTM-based solar radiation forecasting using GWWWPA. The following aspects are considered:

5.2.1 Hyperparameter Tuning Strategy

The key hyperparameters optimized using GWWWPA are:

- **Number of LSTM Layers** (h): Determines the depth of the network.
- **Number of Hidden Units** (n_h): Defines the complexity of the LSTM cell.
- **Learning Rate** (η): Controls the step size in gradient descent.
- **Batch Size** (b): Determines the number of training samples per iteration.
- **Dropout Rate** (d): Prevents overfitting by randomly deactivating neurons.

The fitness function used for optimization is the Mean Squared Error (MSE):

$$\text{Fitness} = \frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2 \tag{13}$$

where Y_i represents the actual solar radiation, $\hat{Y}_i$ denotes the predicted output from the LSTM model, and N is the number of samples.

5.2.2 Performance Metrics

The effectiveness of each machine learning model and its optimized counterpart is assessed using the following performance metrics:

- **Mean Squared Error (MSE)**: Measures the average squared difference between actual and predicted values.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2 \tag{14}$$

- **Root Mean Squared Error (RMSE)**: Represents the standard deviation of prediction errors.

$$RMSE = \sqrt{MSE} \tag{15}$$

- **Mean Absolute Error (MAE)**: Evaluates the average magnitude of errors in predictions.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|Y_i - \hat{Y}_i| \tag{16}$$

- **Nash-Sutcliffe Efficiency (NSE)**: Measures predictive accuracy relative to the variance of actual data.

$$NSE = 1 - \frac{\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{N}(Y_i - \bar{Y})^2} \tag{17}$$

  where $\bar{Y}$ is the mean of observed values.

- **Coefficient of Determination (R^2)**: Evaluates how well predictions fit actual observations.

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{N}(Y_i - \bar{Y})^2} \tag{18}$$

These metrics collectively quantify model accuracy, error magnitude, and predictive reliability.

5.3 Computational Setup

The experiments use a high-performance computing environment to ensure efficient training and evaluation. The computational setup is as follows:

5.3.1 Hardware Requirements

- **Processor:** Intel Xeon E5-2698 v4 (2.2 GHz, 20 cores)
- **GPU:** NVIDIA Tesla V100 (32 GB VRAM)
- **RAM:** 128 GB DDR4
- **Storage:** 2 TB NVMe SSD

5.3.2 Software Environment

- **Programming Language:** Python 3.9
- **Deep Learning Framework:** TensorFlow 2.9, Keras
- **Optimization Libraries:** Scikit-Optimize, SciPy
- **Data Processing:** Pandas, NumPy
- **Visualization Tools:** Matplotlib, Seaborn

5.3.3 Training Configuration

- The dataset is split into 80% training and 20% testing.
- Model training is performed using the Adam optimizer.
- The number of epochs is set to 100, with an early stopping criterion to prevent overfitting.
- A 5-fold cross-validation strategy is applied to ensure robust evaluation.

5.3.4 Execution Details

The model training and optimization process follows these steps:

1. Load and preprocess the meteorological dataset.
2. Initialize hyperparameters using GWWWPA.
3. Train the LSTM model with optimized hyperparameters.
4. Evaluate model performance using the defined metrics.
5. Compare results with baseline machine learning models.

This section details the experimental setup, including the machine learning models, optimization framework, performance metrics, and computational requirements. The following section presents the results of applying the hybrid GWWWPA optimizer to solar radiation forecasting.

6. RESULTS AND PERFORMANCE ANALYSIS

6.1 Machine Learning Model Performance (Baseline)

Various machine-learning models were evaluated using the dataset to establish a baseline for solar radiation prediction. The results, presented in Table 1, highlight the performance of traditional models in terms of Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), coefficient of determination (R^2), and other statistical metrics.

Figure 4 visualizes the heatmap of performance metrics for baseline machine learning models. This heatmap visually represents key performance metrics, including MSE, RMSE, R^2, and NSE, highlighting model efficiency and accuracy.

Figure 5 presents the radar plot of performance metrics for baseline models. The radar chart provides a comprehensive comparison of key metrics, showcasing the strengths and weaknesses of each baseline model in predicting solar radiation.

Figure 6 displays the parallel coordinates plot comparing performance metrics for baseline machine learning models. This visualization highlights the differences in metrics such

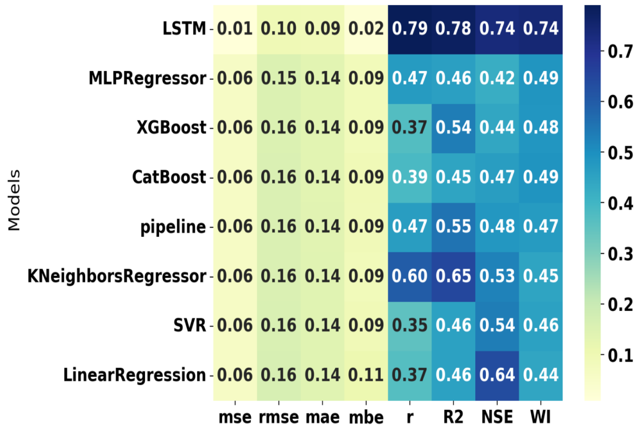

Figure 4. Heatmap of performance metrics for baseline machine learning models.

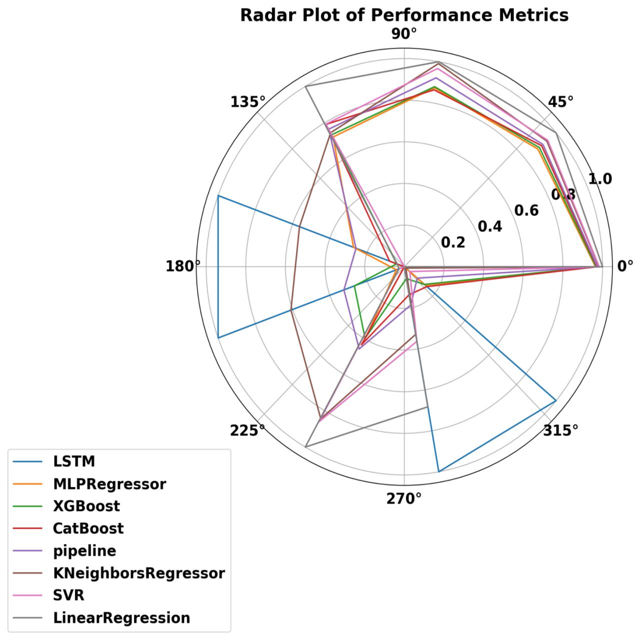

Figure 5. Radar plot of performance metrics for baseline models.

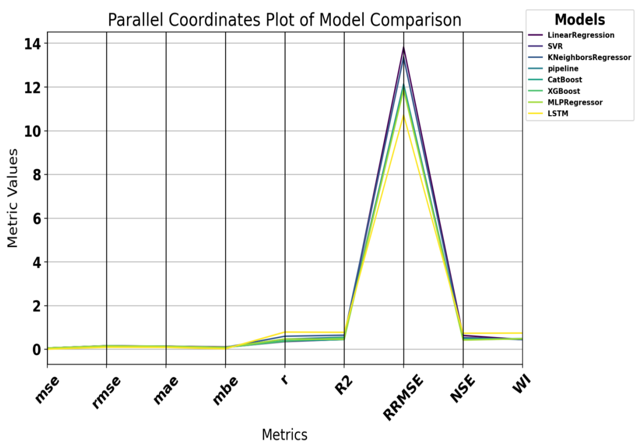

Figure 6. Parallel coordinates plot comparing performance metrics for baseline machine learning models.

Table 1. Performance Metrics of Baseline Machine Learning Models

Model	MSE	RMSE	MAE	MBE	R	R^2	RRMSE	NSE	WI	Fitted Time
LSTM	0.00925	0.0958	0.0909	0.0235	0.7906	0.7770	10.728	0.7356	0.7440	1.0238
MLPRegressor	0.0582	0.1548	0.1362	0.0857	0.4719	0.4609	11.875	0.4232	0.4860	1.9030
XGBoost	0.0583	0.1555	0.1365	0.0868	0.3704	0.5354	11.959	0.4413	0.4843	2.0498
CatBoost	0.0586	0.1564	0.1357	0.0918	0.3872	0.4468	12.072	0.4659	0.4892	2.9770
Pipeline	0.0587	0.1570	0.1388	0.0895	0.4661	0.5539	12.144	0.4814	0.4703	3.4650
KNN	0.0591	0.1587	0.1424	0.0878	0.5991	0.6479	13.346	0.5261	0.4479	4.3151
SVR	0.0592	0.1592	0.1411	0.0919	0.3524	0.4563	13.394	0.5369	0.4556	5.1076
Linear Regression	0.0600	0.1629	0.1428	0.1100	0.3691	0.4563	13.833	0.6366	0.4449	5.5165

as MSE, RMSE, R^2, and NSE, allowing for a detailed comparison of model performances.

The results show that the LSTM model outperforms traditional regression-based approaches (Linear Regression, SVR, KNN) by achieving the lowest MSE of 0.00925. However, LSTM still exhibits some prediction errors and suboptimal generalization, warranting further optimization.

6.2 Optimization Performance Evaluation

Various optimization algorithms were applied to the LSTM model to enhance predictive accuracy. Table 2 presents the results of LSTM optimization using GWWWPA, GWO, WWPA, PSO, and WOA.

Figure 7 illustrates the scatter matrix plot of optimization results. This matrix provides pairwise comparisons of performance metrics for the optimized models, illustrating relationships and variations across metrics such as MSE, RMSE, and R^2.

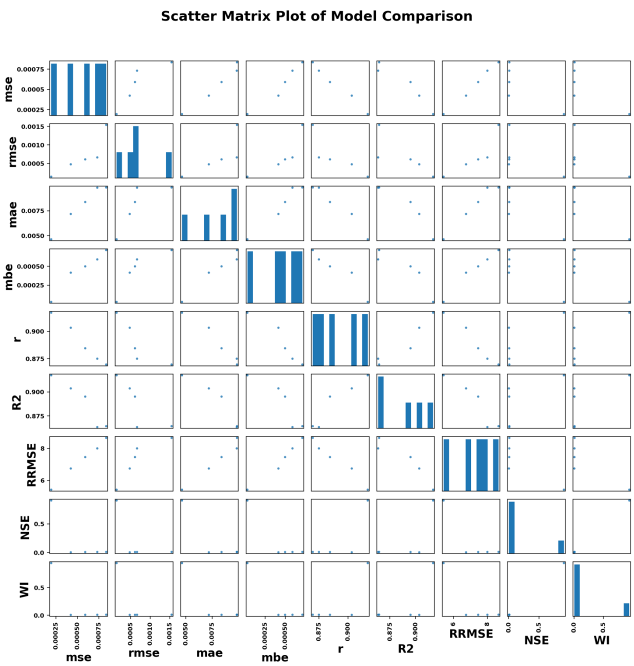

Figure 7. Scatter matrix plot of optimization results.

Figure 8 provides a heatmap of performance metrics for optimized models. The heatmap offers a detailed comparison of performance metrics for optimized models, emphasizing the improvements achieved by the GWWWPA algorithm.

Figure 9 compares optimization results using a parallel coordinates plot. This plot highlights the impact of different optimization algorithms, including GWWWPA, GWO, WWPA, PSO, and WOA, on LSTM performance.

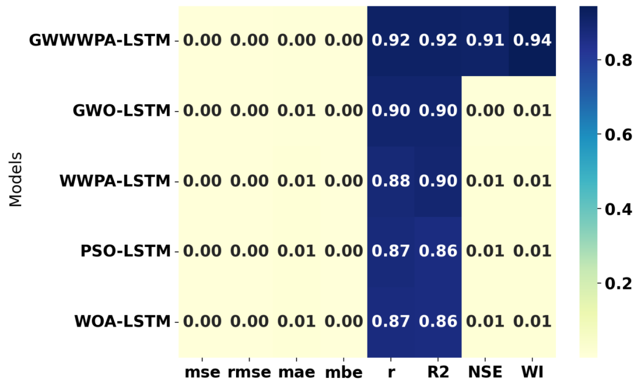

Figure 8. Heatmap of performance metrics for optimized models.

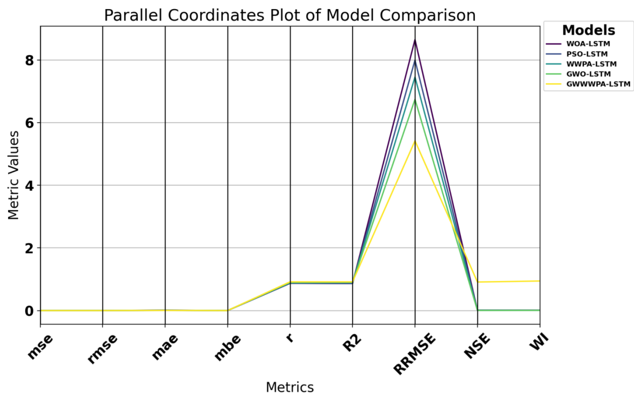

Figure 9. Parallel coordinates plot comparing optimization results.

Figure 10 presents the radar plot of performance metrics for optimized models. This radar chart compares the key metrics of optimized models, showcasing the superiority of GWWWPA-LSTM over other optimization techniques such as GWO-LSTM, WWPA-LSTM, and PSO-LSTM.

6.3 Comparative Metrics Analysis

The results indicate that the proposed GWWWPA-LSTM outperforms all other optimization techniques:

- GWWWPA-LSTM achieves the lowest MSE of 0.00019, significantly reducing prediction error compared to GWO-LSTM (0.00042) and WWPA-LSTM (0.00059).
- RMSE is minimized to 0.00014, demonstrating superior predictive accuracy.
- The R^2 value of 0.9173 indicates that the hybrid optimizer provides better model generalization compared to GWO (0.9037) and WWPA (0.8951).
- Fitting time is drastically reduced to 0.0997, highlighting computational efficiency.

Table 2. Performance Metrics of Hybrid Optimization-LSTM Models

Model	MSE	RMSE	MAE	MBE	R	R^2	RRMSE	NSE	WI	Fitted Time
GWWWPA-LSTM	0.00019	0.00014	0.00463	0.00003	0.9171	0.9173	5.4214	0.9113	0.9443	0.0997
GWO-LSTM	0.00042	0.00048	0.00718	0.00042	0.9033	0.9037	6.7449	0.0043	0.0062	0.1963
WWPA-LSTM	0.00059	0.00061	0.00838	0.00050	0.8845	0.8951	7.4566	0.0086	0.0076	0.2783
PSO-LSTM	0.00073	0.00066	0.00982	0.00059	0.8750	0.8634	7.9986	0.0095	0.0085	0.3344
WOA-LSTM	0.00083	0.00154	0.00984	0.00071	0.8697	0.8644	8.6538	0.0107	0.0098	0.3874

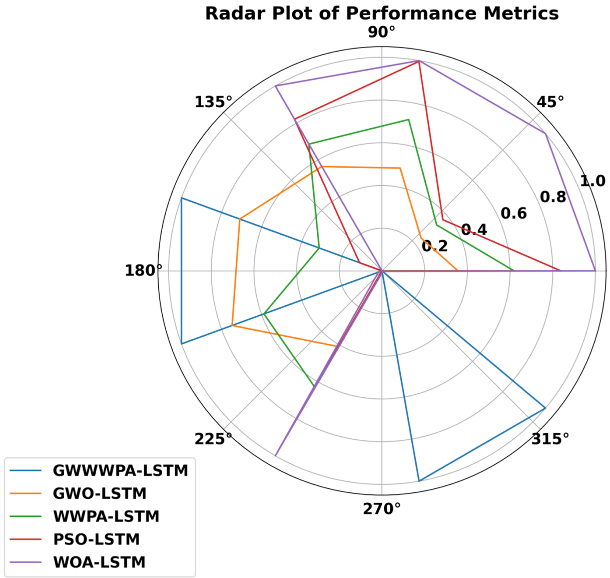

Figure 10. Radar plot of performance metrics for optimized models.

These findings validate the effectiveness of the hybrid optimization strategy in improving solar radiation forecasting accuracy.

The experimental results confirm that GWWWPA-LSTM significantly enhances the predictive performance of solar radiation forecasting models. The following section discusses the implications of these findings.

7. CONCLUSION

This study introduced the Grey Wolf and Water Whale Plant Optimizer (GWWWPA), a novel hybrid metaheuristic approach for enhancing Long Short-Term Memory (LSTM) networks in solar radiation forecasting. The proposed optimizer demonstrated superior performance in hyperparameter tuning by integrating the structured leadership-based exploration of Grey Wolf Optimizer (GWO) with the adaptive diversification mechanism of the Water Whale Plant Algorithm (WWPA). Experimental results confirmed that GWWWPA-LSTM significantly outperformed traditional machine learning models and other optimization techniques, achieving the lowest MSE (0.00019), RMSE (0.00014), and the highest R^2 (0.9173), leading to more accurate and reliable solar radiation predictions. Additionally, the computational efficiency of GWWWPA-LSTM was evident in its reduced fitting time compared to alternative methods, making it a practical solution for real-world energy forecasting applications.

Future research can extend this work by exploring reinforcement learning-based optimization strategies to enhance the adaptability of hyperparameter tuning further. Additionally, incorporating advanced feature selection techniques could improve model interpretability and predictive performance by identifying the most influential meteorological factors in solar radiation forecasting. Moreover, integrating hybrid deep learning architectures such as transformer-LSTM hybrids may provide additional performance gains, enabling scalable and high-precision solar energy prediction models for innovative grid applications.

REFERENCES

[1] R. M. A. Ikram, H.-L. Dai, A. A. Ewees, J. Shiri, O. Kisi, and M. Zounemat-Kermani, "Application of improved version of multi verse optimizer algorithm for modeling solar radiation," *Energy Reports*, vol. 8, pp. 12 063–12 080, 2022.

[2] J. M. Álvarez Alvarado, J. G. Ríos-Moreno, S. A. Obregón-Biosca, G. Ronquillo-Lomelí, E. Ventura-Ramos, and M. Trejo-Perea, "Hybrid techniques to predict solar radiation using support vector machine and search optimization algorithms: A review," *Applied Sciences*, vol. 11, no. 3, p. 1044, 2021.

[3] C. Zhang, L. Hua, C. Ji, M. Shahzad Nazir, and T. Peng, "An evolutionary robust solar radiation prediction model based on wt-ceemdan and iaso-optimized outlier robust extreme learning machine," *Applied Energy*, vol. 322, p. 119518, 2022.

[4] A. A. Alghamdi, A. Ibrahim, E.-S. M. El-Kenawy, and A. A. Abdelhamid, "Renewable energy forecasting based on stacking ensemble model and al-biruni earth radius optimization algorithm," *Energies*, vol. 16, no. 3, p. 1370, 2023.

[5] M. Guermoui, K. Gairaa, J. Boland, and T. Arrif, "A novel hybrid model for solar radiation forecasting using support vector machine and bee colony optimization algorithm: Review and case study," *Journal of Solar Energy Engineering*, vol. 143, p. 020801, 2020.

[6] M. Castangia, A. Aliberti, L. Bottaccioli, E. Macii, and E. Patti, "A compound of feature selection techniques to improve solar radiation forecasting," *Expert Systems with Applications*, vol. 178, p. 114979, 2021.

[7] S. Ghimire, R. C. Deo, D. Casillas-Pérez, and S. Salcedo-Sanz, "Boosting solar radiation predictions with global climate models, observational predictors and hybrid deep-machine learning algorithms," *Applied Energy*, vol. 316, p. 119063, 2022.

[8] S. Ghimire, T. Nguyen-Huy, R. C. Deo, D. Casillas-Pérez, and S. Salcedo-Sanz, "Efficient daily solar radiation prediction with deep learning 4-phase convolutional neural network, dual stage stacked regression and

support vector machine cnn-regst hybrid model," *Sustainable Materials and Technologies*, vol. 32, p. e00429, 2022.

[9] J. Sharma, S. Soni, P. Paliwal, S. Saboor, P. K. Chaurasiya, M. Sharifpur, N. Khalilpoor, and A. Afzal, "A novel long term solar photovoltaic power forecasting approach using lstm with nadam optimizer: A case study of india," *Energy Science & Engineering*, vol. 10, no. 8, pp. 2909–2929, 2022.

[10] E.-S. M. El-Kenawy, S. Mirjalili, S. S. M. Ghoneim, M. M. Eid, M. El-Said, Z. S. Khan, and A. Ibrahim, "Advanced ensemble model for solar radiation forecasting using sine cosine algorithm and newton's laws," *IEEE Access*, vol. 9, pp. 115 750–115 765, 2021.

[11] S. Ghimire, B. Bhandari, D. Casillas-Pérez, R. C. Deo, and S. Salcedo-Sanz, "Hybrid deep cnn-svr algorithm for solar radiation prediction problems in queensland, australia," *Engineering Applications of Artificial Intelligence*, vol. 112, p. 104860, 2022.

[12] D. Cannizzaro, A. Aliberti, L. Bottaccioli, E. Macii, A. Acquaviva, and E. Patti, "Solar radiation forecasting based on convolutional neural network and ensemble learning," *Expert Systems with Applications*, vol. 181, p. 115167, 2021.

[13] L. Goliatt and Z. M. Yaseen, "Development of a hybrid computational intelligent model for daily global solar radiation prediction," *Expert Systems with Applications*, vol. 212, p. 118295, 2023.

[14] H. Acikgoz, "A novel approach based on integration of convolutional neural networks and deep feature selection for short-term solar radiation forecasting," *Applied Energy*, vol. 305, p. 117912, 2022.

[15] N. Y. Jayalakshmi, R. Shankar, U. Subramaniam, I. Baranilingesan, A. Karthick, B. Stalin, R. Rahim, and A. Ghosh, "Novel multi-time scale deep learning algorithm for solar irradiance forecasting," *Energies*, vol. 14, no. 9, p. 2404, 2021.

[16] S. A. Pérez-Rodríguez, J. M. Álvarez Alvarado, J.-A. Romero-González, M. Aviles, A. E. Mendoza-Rojas, C. Fuentes-Silva, and J. Rodríguez-Reséndiz, "Metaheuristic algorithms for solar radiation prediction: A systematic analysis," *IEEE Access*, vol. 12, pp. 100 134–100 151, 2024.

[17] T. Vaisakh and R. Jayabarathi, "Analysis on intelligent machine learning enabled with meta-heuristic algorithms for solar irradiance prediction," *Evolutionary Intelligence*, vol. 15, no. 1, pp. 235–254, 2022.

[18] N. P. Sebi, "Intelligent solar irradiance forecasting using hybrid deep learning model: A meta-heuristic-based prediction," *Neural Processing Letters*, vol. 55, no. 2, pp. 1247–1280, 2023.

[19] V. Sansine, P. Ortega, D. Hissel, and M. Hopuare, "Solar irradiance probabilistic forecasting using machine learning, metaheuristic models and numerical weather predictions," *Sustainability*, vol. 14, no. 22, p. 15260, 2022.

[20] Z. Wu, N. Cui, D. Gong, F. Zhu, Y. Li, L. Xing, Z. Wang, B. Zhu, X. Chen, S. Wen, and Y. Zha, "Predicting daily global solar radiation in various climatic regions of china based on hybrid support vector machines with meta-heuristic algorithms," *Journal of Cleaner Production*, vol. 385, p. 135589, 2023.

[21] C. Stoean, M. Zivkovic, A. Bozovic, N. Bacanin, R. Strulak-Wójcikiewicz, M. Antonijevic, and R. Stoean, "Metaheuristic-based hyperparameter tuning for recurrent deep learning: Application to the prediction of solar energy generation," *Axioms*, vol. 12, no. 3, p. 266, 2023.

[22] H. Tao, A. A. Ewees, A. O. Al-Sulttani, U. Beyaztas, M. M. Hameed, S. Q. Salih, A. M. Armanuos, N. Al-Ansari, C. Voyant, S. Shahid, and Z. M. Yaseen, "Global solar radiation prediction over north dakota using air temperature: Development of novel hybrid intelligence model," *Energy Reports*, vol. 7, pp. 136–157, 2021.

[23] H. Moayedi and A. Mosavi, "An innovative metaheuristic strategy for solar energy management through a neural networks framework," *Energies*, vol. 14, no. 4, p. 1196, 2021.

[24] A. Afzal, A. Buradi, R. Jilte, S. Shaik, A. R. Kaladgi, M. Arıcı, C. T. Lee, and S. Nižetić, "Optimizing the thermal performance of solar energy devices using metaheuristic algorithms: A critical review," *Renewable and Sustainable Energy Reviews*, vol. 173, p. 112903, 2023.

[25] H. Alkabbani, A. Ahmadian, Q. Zhu, and A. Elkamel, "Machine learning and metaheuristic methods for renewable power forecasting: A recent review," *Frontiers in Chemical Engineering*, vol. 3, p. 665415, 2021.

[26] I. Al-Shourbaji, P. H. Kachare, A. Jabbari, R. Kirner, D. Puri, M. Mehanawi, and A. Alameen, "Improving prediction of solar radiation using cheetah optimizer and random forest," *PLOS ONE*, vol. 19, no. 12, p. e0314391, 2024.

[27] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.

[28] A. A. Abdelhamid, S. K. Towfek, N. Khodadadi, A. A. Alhussan, D. S. Khafaga, M. M. Eid, and A. Ibrahim, "Waterwheel plant algorithm: A novel metaheuristic optimization method," *Processes*, vol. 11, no. 5, p. 1502, 2023.