IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A IMPROVED SOLAR PHOTOVOLTAIC POWER FORECASTING USING OPTIMIZED METAHEURISTIC LONG SHORT-TERM MEMORY

[1] Dr. Rajani Rajalingam, [2]Dr. Veguru Gayatri

[1] Professor, [2] Associate Professor

[*]Department of Computer Science and Engineering

Geethanjali Institute of Science and Technology (Autonomous), Nellore, Andhra Pradesh, India

Abstract: Solar photovoltaic (PV) power forecasting plays a vital role in efficient energy management, grid stability, and the large-scale integration of renewable energy sources into modern power systems. The inherent variability of solar power generation, caused by fluctuating weather parameters such as solar irradiance, ambient temperature, module temperature, humidity, and cloud cover, makes accurate prediction a challenging task. Conventional machine learning models and standard Long Short-Term Memory (LSTM) networks often struggle to capture complex nonlinear relationships present in solar data and additionally require extensive manual hyperparameter tuning, which reduces their efficiency and scalability. This paper proposes an Optimized Metaheuristic LSTM model that integrates the temporal learning capability of LSTM networks with metaheuristic optimization for automatic hyperparameter tuning. Historical solar and weather data are pre-processed through cleaning, normalization, and feature selection before being transformed into time-series sequences suitable for sequential learning. A metaheuristic optimization process iteratively tunes parameters such as the learning rate, number of neurons, batch size, and number of epochs based on validation performance, thereby minimizing manual intervention. The proposed model is implemented as an interactive Streamlit-based web application and is evaluated using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and the R^2 score. Experimental results show that the Optimized Metaheuristic LSTM achieves an RMSE of 2.10, an MAE of 1.75, an MAPE of 6.3%, and an R^2 score of 0.96, outperforming Linear Regression, Artificial Neural Network (ANN), standard LSTM, Deep LSTM, and CNN-LSTM models, confirming its effectiveness for reliable and near real-time solar power forecasting.

Index Terms - Solar Photovoltaic Power Forecasting, Long Short-Term Memory, Metaheuristic Optimization, Deep Learning, Renewable Energy, Time-Series Analysis, Hyperparameter Tuning.

I. INTRODUCTION

Solar photovoltaic (PV) energy has become a vital component of modern power systems owing to its renewable and eco-friendly nature. With the growing global demand for clean energy, solar power plays a key role in reducing dependence on fossil fuels and minimizing environmental impact. However, solar power generation is highly dependent on environmental factors such as solar irradiance, temperature, humidity, and cloud cover, which makes it inherently variable and uncertain. This variability creates significant challenges in accurately forecasting solar energy output and motivates the development of intelligent forecasting techniques.

Accurate solar power forecasting is essential for efficient energy management, grid stability, and optimal utilization of renewable resources. The increasing global demand for clean and sustainable energy, together with the rapid growth of smart grids and renewable energy integration, has increased the need for intelligent forecasting systems that can adapt to real-time changes in environmental conditions. Accurate solar power prediction helps in better load scheduling, efficient utilization of energy resources, and minimizing the mismatch between power generation and demand. Without reliable forecasting, energy providers may face issues such as overproduction, energy wastage, or power shortages, which directly affect economic and operational efficiency.

Traditional forecasting methods and basic machine learning models are often inadequate in capturing the complex and nonlinear patterns present in solar data [1], [2]. Deep learning models such as LSTM are well-suited for sequential data modeling and can learn long-term temporal dependencies; however, they require manual tuning of parameters such as learning rate, batch size, and number of neurons, which is time-consuming and may lead to suboptimal configurations [3], [4]. The advancement of artificial intelligence and optimization techniques provides an opportunity to enhance traditional forecasting approaches: by combining deep learning models with metaheuristic optimization, it becomes possible to automatically identify optimal model parameters and improve performance without manual intervention, while also ensuring scalability and robustness of the system.

This paper proposes an Optimized Metaheuristic LSTM model that analyzes historical solar and weather data to forecast solar photovoltaic power output. The remainder of this paper is organized as follows: Section II presents a literature survey of related work; Section III describes the proposed methodology; Section IV presents implementation details; Section V discusses results; and Section VI concludes with future scope.

II. LITERATURE SURVEY

Solar power forecasting has been studied extensively using statistical methods, machine learning algorithms, and deep learning techniques. Traditional approaches such as linear regression assume a linear relationship between input features such as solar irradiance and temperature and the resulting power output; while simple, fast, and computationally efficient, such models fail to handle the nonlinear relationships and complex variations present in solar data caused by changing weather conditions [1].

Table 1: Existing Solar Power Forecasting Approaches

Model	Description	Limitations
Linear Regression	Models a linear relationship between weather parameters and power output.	Cannot capture nonlinear relationships.
Artificial Neural Network (ANN)	Feedforward network modeling simple nonlinear relationships.	Struggles with long-term dependencies; prone to overfitting.
Standard LSTM	Recurrent network capturing temporal dependencies in time-series data.	Requires extensive manual hyperparameter tuning.
CNN-LSTM (Hybrid)	Combines CNN-based spatial feature extraction with LSTM temporal learning.	High computational complexity and longer training time.

Artificial Neural Networks (ANN) have been applied to model nonlinear relationships between weather parameters such as temperature, humidity, and solar radiation and power output [4]. While ANN models provide reasonable accuracy for small datasets, they struggle to capture long-term dependencies in time-series data and are prone to overfitting when dealing with complex and large datasets, with performance highly sensitive to network architecture and parameter selection.

Long Short-Term Memory (LSTM) networks have been widely adopted for solar power forecasting due to their ability to capture sequential patterns and temporal relationships in historical solar irradiance, temperature, and power output data [3], [6]. Gensler et al. proposed a deep learning approach combining autoencoders with LSTM networks for solar power forecasting [3]. Wang et al. studied short-term solar power forecasting based on LSTM neural networks and reported improved accuracy over conventional machine learning models [6]. Hybrid models such as Convolutional LSTM (ConvLSTM), originally proposed by Shi et al. for precipitation nowcasting, combine convolutional operations with LSTM to jointly capture spatial and temporal patterns [5], an idea that has since been adapted to other time-series forecasting hybrids such as CNN-LSTM for solar power prediction.

Across these studies, a recurring limitation is that deep learning models such as LSTM and CNN-LSTM require careful manual hyperparameter tuning, including learning rate, batch size, number of neurons, and epochs, which is time-consuming and may lead to suboptimal configurations [2]. Existing systems also show limited scalability when applied to large and complex datasets. These research gaps motivate the present work, which integrates metaheuristic optimization with LSTM to automate hyperparameter tuning and improve forecasting accuracy.

III. PROPOSED METHODOLOGY

The proposed system forecasts solar photovoltaic power output using an Optimized Metaheuristic Long Short-Term Memory model. The methodology involves data collection, data preprocessing, feature selection, sequence generation, model training with metaheuristic-based hyperparameter optimization, and performance evaluation.

A. Data Collection and Dataset Description

The dataset consists of historical solar and weather parameters collected at regular time intervals, including solar irradiance, ambient temperature, module temperature, humidity, time-based features such as time of day and day of year, and historical power output values. This data forms the foundation for training the forecasting model.

B. Data Preprocessing

Preprocessing is one of the most important stages in the pipeline and ensures the quality and usability of data for training and evaluation. Data processing involves cleaning the dataset by handling missing values, correcting inconsistent data, and detecting outliers, along with normalization or scaling of numerical features. Data visualization techniques such as histograms, scatter plots, and correlation matrices are used to understand feature distributions, trends, seasonal patterns, and dependencies between weather parameters and power output. Feature engineering transforms time-based attributes such as date and time into meaningful numerical formats, such as hour, day, or cyclic features, to better capture temporal behavior.

C. Feature Selection

Feature selection reduces dimensionality and improves model efficiency by retaining only the most relevant attributes that influence solar power generation, including solar irradiance, ambient temperature, module temperature, and time-based features. This step helps reduce overfitting, lowers computational cost, and improves prediction accuracy.

D. Data Splitting and Sequence Generation

Once the data is cleaned and relevant features are selected, it is transformed into time-series sequences suitable for LSTM input and divided into training and testing sets, using a split ratio such as 80:20, to evaluate how well the model generalizes to unseen data.

E. Model Architecture: LSTM and Optimized Metaheuristic LSTM

LSTM is a type of Recurrent Neural Network (RNN) designed to handle time-series data and learn long-term dependencies. The LSTM model consists of memory cells and three main gates - input, forget, and output gates - that control the flow of information, enabling the network to retain important information and discard irrelevant data. This makes LSTM effective at handling the nonlinear and complex relationships present in solar power data, capturing both short-term fluctuations and long-term seasonal dependencies.

To further improve performance, a metaheuristic optimization technique is integrated with the LSTM model for automatic hyperparameter tuning. The optimization process maintains a population of candidate solutions, where each solution represents a combination of hyperparameters such as learning rate, batch size, and number of neurons. A fitness function evaluates each candidate based on model performance using validation metrics such as RMSE or MAE. The best-performing candidates are selected, and new candidate solutions are generated by modifying existing ones to explore better configurations. This selection-and-update process is repeated iteratively until the best parameter combination is found, forming an optimization loop in which the model is continuously trained, validated, and improved. This approach reduces manual tuning effort and improves prediction accuracy and generalization.

F. Hyperparameters and Training Configuration

Table 2: Hyperparameters Used

Parameter	Default Value	Tuned By
Learning Rate	0.001	Metaheuristic optimization
Batch Size	64	Metaheuristic optimization
Number of Neurons	50	Metaheuristic optimization
Number of Epochs	100	Metaheuristic optimization

The model is trained with an initial set of randomly selected hyperparameters, and the metaheuristic optimization algorithm iteratively updates these parameters based on validation performance until the best-performing configuration is obtained.

IV. IMPLEMENTATION

A. Software and Hardware Requirements

The system is implemented using the Python programming language. The deep learning model is built using TensorFlow with the Keras high-level API, which simplifies the construction and training of LSTM networks. Data processing is handled using NumPy for numerical computation, Pandas for dataset manipulation, and Scikit-learn for preprocessing and evaluation metrics, while Matplotlib is used for visualization. The application is developed in Visual Studio Code, and a Streamlit-based web interface is used as the front end. The system was developed using a standard computing configuration with an Intel i5 processor, 8 GB RAM, and 256 GB storage, running on the Windows operating system, with the dataset stored in CSV format.

B. System Architecture

The overall system architecture consists of the following sequential stages: (i) Dataset, the historical solar and weather data input layer; (ii) Data Split, dividing the dataset into training and testing partitions; (iii) an Optimization Loop comprising Train LSTM, Validate, and Update Parameters steps that are repeated iteratively; (iv) Best Model selection, where the configuration with the lowest validation error is retained; and (v) Final Testing, in which the optimized model is evaluated on unseen test data before deployment for prediction.

C. Web Application

A user-friendly web-based interface is developed using the Streamlit framework, allowing users to navigate through dataset information, data preprocessing, model training, and prediction modules. The interface enables users to view dataset details, run predictions on test samples or on their own uploaded CSV data, and visualize actual versus predicted power values along with the absolute prediction error in real time, supporting practical and interactive solar power forecasting.

D. Testing

A comprehensive testing strategy, comprising module testing, integration testing, and acceptance testing, was used to validate the system. Table 3 summarizes representative test cases and their outcomes.

Table 3: Representative Test Cases and Results

Test ID	Test Scenario	Expected Result	Status
TC01	Dataset upload	Dataset uploaded successfully	Pass
TC02	Data preprocessing	Data cleaned and normalized	Pass
TC03	Sequence generation	Sequences generated correctly	Pass
TC04	Model training	Model trained successfully	Pass
TC05	Hyperparameter optimization	Parameters optimized successfully	Pass
TC06	Prediction	Power output predicted correctly	Pass

V. RESULTS AND DISCUSSION

The proposed Optimized Metaheuristic LSTM model was evaluated on the solar PV dataset using standard regression metrics: RMSE, MAE, MAPE, and the R^2 score. The preprocessing pipeline successfully cleaned the dataset and produced consistent, normalized time-series sequences, which were validated through the module and integration tests described in Section IV-D.

The model was trained on the training partition and evaluated on the held-out test partition. The optimization loop, consisting of repeated training, validation, and parameter updates, progressively reduced validation error until the best-performing configuration was obtained. Table 4 compares the performance of the proposed Optimized Metaheuristic LSTM against Linear Regression, ANN, standard LSTM, Deep LSTM, and CNN-LSTM models.

Table 4: Performance Evaluation of Forecasting Models

Model	RMSE	MAE	MAPE (%)	R^2
Linear Regression	5.82	4.75	18.6	0.78
ANN	4.35	3.60	14.2	0.85
LSTM	3.12	2.45	9.8	0.91
Deep LSTM	2.85	2.20	8.7	0.93
CNN-LSTM	2.60	2.05	7.9	0.94
Optimized LSTM (Proposed)	2.10	1.75	6.3	0.96

As shown in Table 4, the proposed Optimized Metaheuristic LSTM model achieves the lowest RMSE (2.10), the lowest MAE (1.75), the lowest MAPE (6.3%), and the highest R^2 score (0.96) among all compared models. The Linear Regression model performs the weakest, confirming its inability to capture the nonlinear and dynamic behavior of solar power data. The ANN model offers some improvement by modeling nonlinear relationships but still falls short of LSTM-based approaches in capturing long-term temporal dependencies. The standard LSTM, Deep LSTM, and CNN-LSTM models show progressively lower error and higher R^2 values, demonstrating the benefit of temporal and, in the case of CNN-LSTM, joint spatial-temporal feature learning. The proposed model further improves upon these results by automatically tuning critical hyperparameters through metaheuristic optimization rather than relying on manually selected values, leading to a more suitable configuration for the dataset and consequently lower prediction error.

These results confirm that combining LSTM-based temporal learning with metaheuristic-driven hyperparameter optimization yields a more accurate and reliable forecasting model than traditional and standalone deep learning approaches, while also reducing the manual effort typically required for parameter tuning. The accompanying Streamlit interface further supports near real-time prediction and visualization of actual versus predicted power values, which is useful for practical solar energy management and grid stability applications.

VI. CONCLUSION

This paper presented an Optimized Metaheuristic Long Short-Term Memory model for solar photovoltaic power forecasting, implemented as an interactive Streamlit web application. By combining LSTM-based temporal learning with metaheuristic optimization for automatic hyperparameter tuning, the proposed system reduces manual tuning effort while improving prediction accuracy. Data preprocessing steps including cleaning, normalization, feature engineering, and feature selection improved data quality and model training efficiency. Experimental results show that the proposed model achieves an RMSE of 2.10, an MAE of 1.75, an MAPE of 6.3%, and an R^2 score of 0.96, outperforming Linear Regression, ANN, standard LSTM, Deep LSTM, and CNN-LSTM models. The developed system contributes to more accurate and timely solar power forecasts, supporting better energy management and grid stability. Future work will focus on incorporating additional weather factors such as cloud cover, integrating the system with smart grids, expanding the training dataset to further improve accuracy, developing a mobile application for accessible monitoring, and enhancing the web interface with advanced dashboards and visualizations.

REFERENCES

[1] H. Yang, Z. Dong, and Z. Liu, "A review of solar power forecasting methods," Renewable and Sustainable Energy Reviews, vol. 81, pp. 782–797, 2018.

[2] J. Voyant, G. Notton, S. Kalogirou, et al., "Machine learning methods for solar radiation forecasting: A review," Renewable Energy, vol. 105, pp. 569–582, 2017.

[3] A. Gensler, J. Henze, B. Sick, and N. Raabe, "Deep learning for solar power forecasting – an approach using autoencoder and LSTM neural networks," in Proc. IEEE Int. Conf. Systems, Man, and Cybernetics, 2016, pp. 2858–2865.

[4] Y. Wang, Q. Chen, et al., "Short-term solar power forecasting based on LSTM neural network," IEEE Access, vol. 7, pp. 125412–125421, 2019.

[5] X. Shi, Z. Chen, H. Wang, et al., "Convolutional LSTM network: a machine learning approach for precipitation nowcasting," in Advances in Neural Information Processing Systems, 2015.

[6] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning. New York: Springer, 2009.

[7] Kaggle, "Solar power generation dataset," Available: https://www.kaggle.com/ (accessed for dataset).

[8] Scikit-learn Documentation, "Machine learning in Python," Available: https://scikit-learn.org/.

[9] TensorFlow Documentation, “An end-to-end open source machine learning platform,” Available: https://www.tensorflow.org/.

[10] F. Chollet, Deep Learning with Python. Shelter Island, NY: Manning Publications, 2017.

[11] D. P. Kingma and J. Ba, “Adam: a method for stochastic optimization,” in Proc. Int. Conf. Learning Representations (ICLR), 2015.

.” [Online]. Available: https://www.kaggle.com/