Journal of Operation and Automation in Power Engineering

Vol. XX, No. XX, Dec. 2022, Pages: XXXX (Proofed)

http://joape.uma.ac.ir

Research Paper

Next-Gen Solar Forecasting: PSO-Optimized Bayesian LSTM for Enhanced Accuracy

S.J. Sadheesh Kumar * and *K. Navin Sam*

Department of Electrical and Electronics Engineering, National Institute of Technology Puducherry, Karaikal, India.

Abstract— *Accurate solar photovoltaic power (SPVP) generation forecasting is vital for integrating solar energy into the power grid. This paper presents an advanced forecasting model using a Bayesian enhanced Long Short-Term Memory neural network (BLSTM NN) model optimized by the Particle Swarm Optimization (PSO) algorithm to elevate the accuracy and reliability of SPVP generation forecasting. The hyperparameters of the BLSTM NN are optimized using the PSO algorithm, resulting in improved forecasting performance. The model is evaluated using a comprehensive dataset comprising five years of historical data from a 1 MW SPVP plant in South India, sampled at 15-minute intervals. Key performance indicators, including Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and R-squared, are used for various analyses. The performance of the proposed model is evaluated through accuracy, uncertainty, scalability, and sensitivity analyses. Experimental results highlight a 16.02% reduction in RMSE, a 22.84% reduction in MAPE, and a 24% improvement in R^2 over the conventional baseline DB model. The outcomes underscore the capability of the proposed model to deliver superior forecasting accuracy. The approach helps integrate reliable and efficient solar power into grid planning.*

Keywords—Bayesian LSTM NN model, forecasting, hyperparameter optimization, particle swarm optimization, solar photovoltaic power, uncertainty quantification.

1. INTRODUCTION

Renewable energy sources are increasingly important in addressing global energy demands and mitigating climate change impacts. Solar photovoltaic power (SPVP) is the primary solution for replacing conventional sources due to availability [1]. Accurate SPVP generation forecasting is vital for integrating SPVP into the grids [2]. The forecasting SPVP generation is challenging due to the inherent variable weather conditions. The variable weather conditions lead to variations in SPVP generation output [3]. The variability causes inaccuracies in forecasting SPVP generation, causing suboptimal power grid planning. Therefore, developing robust forecasting models that accurately capture and forecast the variations is essential.

Time series forecasting is vital in various key areas, including finance and stock marketing, weather and climate forecasting, and energy and demand management. Statistical methods, such as Autoregressive Integrated Moving Average (ARIMA), Seasonal ARIMA (SARIMA), Exponential Smoothing Methods (ETS Models), Vector Autoregression (VAR), Vector Autoregressive Moving Average (VARMA), and Gaussian Process Regression (GPR) are widely used for their simplicity and effectiveness in modelling linear relationships within data [4]. The statistical models struggle with non-linearity, high-dimensional data, long-term forecasting, adaptability, and scalability [5]. In response to the limitations, Artificial Neural Networks (ANN) have gained popularity for their ability to model non-linear relationships. The time-series forecasting model significantly improves performance using feedforward ANNs [6]. The ANNs cannot handle sequential data efficiently because they do not account for temporal dependencies in data points. Then, the Recurrent Neural Networks (RNN) came into the picture to overcome the limitations of traditional ANNs in time series forecasting [7]. Initially, the long-short-term memory neural networks (LSTM NN) models are designed to capture longstanding dependencies in time series data through gates that regulate the flow of information. This architecture allows LSTM NN models to retain relevant past information over extended sequences, making them particularly effective for time series forecasting problems, including time series SPVP generation forecasting [8]. However, LSTM NN models often provide deterministic forecasting but fail to forecast uncertainty inherent in real-world data [9].

The issue of uncertainty forecasting in time series data is addressed by the Bayesian concept in machine learning [10]. By treating model parameters as probabilistic entities rather than fixed values, Bayesian-based models can provide point estimates and confidence intervals, thereby offering a measure of uncertainty associated with forecasting [11]. Hybridising the LSTM NN model and Bayesian concepts into one model can capture longstanding dependencies and uncertainties in the time series data. By incorporating Bayesian inference, the Bayesian-enhanced LSTM (BLSTM NN) extends the traditional LSTM NN model by integrating Bayesian inference into the network architecture [12]. The integration allows for estimating posterior distributions over model parameters, resulting in forecasting with a quantifiable measure of uncertainty. In SPVP generation forecasting, BLSTM NN models can provide more robust and reliable forecasting

Received: 13 Dec. 2024
Revised: 18 Apr. 2025
Accepted: 30 Apr. 2025
*Corresponding author:
E-mail: sjs.nitpy@gmail.com (S.J. Sadheesh Kumar)
DOI: 10.22098/joape.2025.16360.2266

by accounting for variability and unforecastable weather-related inputs. The probabilistic approach enhances the interpretability and trustworthiness of the model. However, the hyperparameters of the model must be optimized to acquire the maximum ability. Without hyperparameter optimization, the BLSTM NN model may fail to provide accurate, reliable, and efficient forecasts, making it unsuitable for real-world applications [13].

Optimization algorithms such as grid Search, random search and Bayesian Optimization (BO) are generally used in neural network models to optimize hyperparameters due to their simplicity [14]. They are slow, computationally expensive, struggle with high-dimensional spaces, and may find suboptimal solutions. The widely accepted Particle Swarm Optimization (PSO) algorithm addresses the limitations. The Particle Swarm Optimization (PSO) algorithm is better for balancing exploration-exploitation, accelerating convergence and finding optimal hyperparameters with lower computational cost and easy implementation [15]. However, the PSO algorithm struggles with the issue of high-dimensional search space. Sensitivity analyses are carried out to address the issue of dimensionality reduction and utilise other sound qualities of the PSO algorithm [16]. The dimensions are fixed to suit the objective problem and the PSO algorithm. The proposed forecasting model integrating the PSO algorithm with BLSTM NN models offers a synergistic approach that leverages the strengths of both techniques. The BLSTM NN model provides a framework for uncertainty quantification. The PSO algorithm ensures the hyperparameters are optimally set. The combined approach enhances accuracy and reliability in SPVP generation forecasting applications. Integrating Bayesian concepts and the PSO algorithm represents a significant advancement, offering robust, reliable, and interpretable forecasts essential for effective power grid planning. The significant contributions of the work are listed below.

- The Bayesian-enhanced LSTM NN model optimized by Particle Swarm Optimization for SPVP generation forecasting is proposed.
- The sensitivity analyses based on reduced dimensional hyperparameters of the BLSTM NN model are proposed to be optimized using the PSO algorithm.
- The state-of-the-art baseline models are compared in all aspects with the real-time input dataset to claim the superiority of the proposed model.
- The thorough performance analyses based on accuracy, uncertainty, scalability, and sensitivity are conducted for robust model evaluation.

The paper is organised into several key sections. Section 2 outlines the methodology of the proposed PSO-BLSTM NN, detailing the integration of the PSO algorithm with the BLSTM NN model. Section 3 covers experimental setup and validation, including dataset usage, pre-processing, training parameters, and validation techniques. Section 4 contains results and discussions, including various analyses with graphical and tabulated results. The Conclusion in Section 5 summarises the key findings, achievements, and future work to suggest further research directions. References cited are included at the end.

2. Methodology

The concept of Bayesian is used to enhance the LSTM NN model, and the PSO algorithm optimizes the impact full hyperparameters. In the PSO-BLSTM NN model, Bayesian learning and the PSO algorithm are complementary in enhancing forecasting accuracy and reliability [17]. Bayesian learning introduces probabilistic weight estimation, enabling the model to quantify uncertainty and provide confidence intervals, making forecasting more robust, especially under variable climatic conditions [18]. It mitigates overfitting by incorporating prior distributions and refining weight updates through variational inference. On the other hand, the PSO algorithm serves as an efficient global optimizer, optimising key hyperparameters. By balancing exploration and exploitation, the PSO algorithm ensures that the model converges to an optimal configuration, improving training stability and generalisation. Bayesian inference enhances forecasting reliability, while the PSO algorithm optimizes model efficiency and performance, resulting in a highly accurate and adaptive forecasting model for solar PV power forecasting. The LSTM NN cell is the core building block in the proposed forecasting model. In the following section, the LSTM NN cell is discussed in detail.

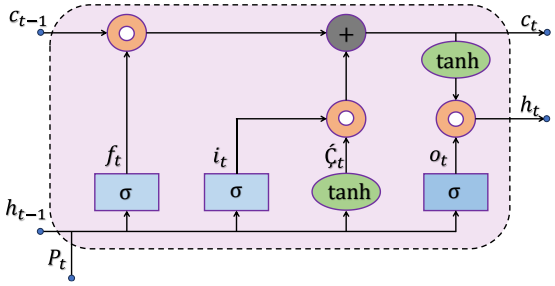

Fig. 1. Standard LSTM NN cell.

2.1. LSTM NN cell

The standard LSTM NN model is a specialised RNN architecture [19]. The Standard LSTM NN cell is shown in Fig. 1 and is designed to effectively model sequential dependencies in time-series data. At first, the historical SPVP is represented as P_t to describe the mathematical framework of the proposed model. Then, the LSTM NN model incorporates gates, viz., input, forget, and output gates, to control the flow of information, mitigating the issue of vanishing gradients and capturing long-term dependencies. In the PSO-BLSTM NN model, the LSTM NN model is the core time-series forecasting engine, extracting temporal patterns from historical SPVP data. The gated mechanisms of the LSTM NN model regulate the information flow, as shown in Fig. 1. The forget gate decides how much of the previous cell state should be retained or discarded based on the previous hidden state and the current input. This is controlled by a sigmoid activation function, which outputs values between 0 (forget completely) and 1 (retain completely). The equations of the gates can be realised from Fig. 1; they are given in the sequence. The forget gate f_t is shown in Eq. (1).

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

where, W_f and b_f are the weight and bias for the forget gate, σ is the sigmoid activation function, h_{t-1} is the previous hidden state, x_t is the current input. The input gate is next, determining how much new information from the current input should be added to the cell state. It consists of two components. The first is a sigmoid function that decides how much information to update. A tanh function that creates a candidate value for updating the cell state. The input gate i_t and candidate cell state $\tilde{C}_t$ are given in Eqs. (2) and (3):

$$i_t = \sigma(W_i \cdot [h_{t-1}, P_t] + b_i) \tag{2}$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, P_t] + b_c) \tag{3}$$

where, W_i, W_c are weight and b_i, b_c are bias terms for the input gate and candidate cell state. The cell state is updated using both the forget gate and input gate. The cell state C_t is given in Eq. (4):

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{4}$$

where, $f_t.C_{t-1}$ determines how much of the previous memory is retained, $i_t.\tilde{C}_t$ adds new information to the cell state. The output gate regulates what portion of the cell state should be exposed as the hidden state at the current time step. The output gate o_t is given in Eq. (5):

$$o_t = \sigma(W_o \cdot [h_{t-1}, P_t] + b_o) \tag{5}$$

where, W_o and b_o are weight and bias terms for the output gate, the activation function σ ensures the output regulation between 0 and 1. Finally, the hidden state is computed using the output gate activation and the updated cell state. The hidden state h_t is given in Eq. (6):

$$h_t = o_t \cdot \tanh(C_t) \tag{6}$$

The flow of information in the LSTM NN cell is understood using the equations. The proposed model introduces the probability distribution for the weights and biases discussed in the above equations through the Bayesian concept to estimate uncertainty in forecasting. The Bayesian concept is discussed in the following section.

2.2. Bayesian enhancement in the LSTM NN model

Integrating Bayesian learning in the LSTM NN model enhances the forecasting reliability and uncertainty quantification [20]. In the BLSTM NN model, the Bayesian concept is applied to LSTM NN model weights using variational inference (VI), ensuring the model produces accurate point forecasting and provides credible intervals [21]. The Bayesian concept also performs regularising, reducing overfitting, especially when trained on limited or highly fluctuating datasets, such as SPVP generation forecasting. The Bayesian LSTM NN model assumes distribution over parameters as given in Eq. (7):

$$P(W, b \mid D) = \frac{P(D \mid W, b)\, P(W, b)}{P(D)} \tag{7}$$

where, D is the data, W, b are weights and biases, respectively, and $P(W, b|D)$ is the posterior distribution of the weights and biases for the given data. $P(D|W, b)$ is the likelihood of the data given the parameters, $P(W, b)$ is the prior distribution over the weights, $P(D)$ is the normalisation term. The model does not learn a fixed W but instead learns a probabilistic distribution over weights, allowing it to estimate uncertainty in forecasting. The stochastic weight updates through VI; since direct Bayesian inference is intractable, VI approximates the posterior as given in Eq. (8):

$$Q(W, b) \approx P(W, b \mid D) \tag{8}$$

where $Q(W, b)$ is a variational distribution for Bayesian inference, and the objective function is to minimize the Kullback-Leibler (KL) divergence between the true posterior and the approximated distribution, as given in Eq. (9):

$$\begin{aligned} L_{\text{VI}} = \mathbb{E}_{Q(W,b)}[\log P(D \mid W, b)] - \\ \text{KL}(Q(W, b) \parallel P(W, b)) \end{aligned} \tag{9}$$

where L_{VI} is Variational loss function, $\mathbb{E}_(Q(W, b))$ is an expectation taken over the approximate posterior distribution $Q(W, b)$ representing the uncertainty-aware learned parameters, and $KL(Q(W, b)P(W, b))$ is KL divergence. The first term of the loss function represents the log-likelihood of data, ensuring model accuracy, and then KL divergence, acting as a regularisation term to prevent overfitting. Thus, the Bayesian LSTM NN model introduces regularisation and uncertainty quantification through probabilistic weight estimation.

2.3. Hyperparameter optimization using PSO algorithm

The BLSTM NN model optimises the hyperparameters using the PSO algorithm for SPVP generation forecasting. Conventional methods such as grid search and random search are computationally demanding for complex problems and may not consistently identify the optimal parameter settings [22]. Then, the PSO algorithm, a swarm-based metaheuristic optimization algorithm, effectively explores the hyperparameter space by modelling the movement of particles within a solution landscape. However, optimizing many hyperparameters leads to a high-dimensional search space, increasing computational complexity and the risk of premature convergence. To effectively handle this issue, a preliminary sensitivity analysis is conducted to identify the most influential hyperparameters affecting model performance [23]. This method sets less impactful parameters at default values, reducing the adequate search space while maintaining optimization efficiency. In the PSO algorithm, each particle signifies a potential hyperparameter set, and its position is updated iteratively based on its personal best performance (p-best) and the global best performance (g-best), maintaining a balanced trade-off between exploration and exploitation. In this framework, the PSO algorithm optimizes critical BLSTM NN model hyperparameters. This improves training stability, faster convergence, and enhanced forecasting accuracy. The sensitivity analyses are reported with the essence of its results to understand the impactful parameters of BLSTM in Table 1. The dimensionality reduction by sensitivity analysis of the PSO algorithm allows the model to efficiently navigate high-dimensional search spaces, reducing computational costs while achieving superior performance. By leveraging the PSO algorithm, the PSO-BLSTM NN model outperforms conventional models, demonstrating improved forecast accuracy, robustness, and generalisation in SPVP generation forecasting. The most influential hyperparameters of the BLSTM NN model are the Learning rate (η), Number of LSTM NN unit (n_h), Number of layers (l), Dropout rate (p), and Batch size (B) according to the sensitivity analysis shown in Table 1.

Table 1. Essence of hyperparameter sensitivity analyses report.

Hyperparameter	Tested range	Optimal range
Learning rate (η)	0.0001 - 0.01	0.0005 - 0.005
Number of LSTM units (n_h)	16 - 256	32 - 128
Number of layers (l)	1 to 15 layers	2 to 11 layers
Dropout rate (p)	0.05 - 0.5	0.1 - 0.3
Batch size (B)	16 - 128	32 - 64

These parameters define the solution space θ. Each particle in the PSO algorithm represents a candidate solution θ_i. The particles are updated iteratively based on their velocities as given in Eqs. (10) to (12):

$$\theta_i = (n_h, p, \eta, B, l)_i \tag{10}$$

$$\begin{aligned} v_{i+1} = \omega v_i + c_1 r_1 (p_{\text{best},i} - \theta_i) + \\ c_2 r_2 (g_{\text{best}} - \theta_i) \end{aligned} \tag{11}$$

$$\theta_{i+1} = \theta_i + v_{i+1} \tag{12}$$

where, v_i represents the velocity of the particle at iteration i, while ω is the inertia weight that regulates the balance between

exploration and exploitation. The parameters c_1, c_2 are cognitive and social coefficients, respectively, r_1, r_2 are randomly generated numbers in the range [0,1] for stochasticity. In addition, $p_{best,i}$ is the best-known position of the particle, and g_{best} represents the global optimal position. The model error function minimised by the PSO algorithm is given as in Eq. (13):

$$\min_{\theta} L(\theta) = \frac{1}{N}\sum_{j=1}^{N}(y_j - \hat{y}_j)^2 \tag{13}$$

where y_j is the actual SPVP generated and $\hat{y}_j$ is the forecasted SPVP-generated output. The convergence criteria is given in Eq. (14):

$$|g_{\text{best},t} - g_{\text{best},t-1}| < \varepsilon \tag{14}$$

a small three should script epsilon, ensuring convergence and the condition of the PSO algorithm to terminate. With a clear understanding of the PSO algorithm and optimisation of hyperparameters through iterations, the integration of the PSO algorithm to BLSTM will be discussed in the following section.

2.4. Integration of PSO algorithm in BLSTM NN model

The Bayesian loss function L_{VI} is integrated with the PSO algorithm to combine the benefits of probabilistic forecasting and efficient hyperparameter optimization. The Bayesian loss function, derived from VI, ensures that the model estimates uncertainty by learning a distribution over weights rather than single-point values, thereby improving generalisation and robustness. Meanwhile, the PSO algorithm optimizes critical hyperparameters, ensuring that the model achieves optimal convergence and improved forecasting accuracy. The final objective function minimizes the forecasting errors while maintaining uncertainty quantification, leading to a highly accurate and stable PSO-BLSTM NN model for SPVP forecasting. The final objective is given as:

$$\begin{aligned} L_{\text{VI}} = \min_{\theta}\mathbb{E}_{Q(W,b)}[\log P(D \mid W,b)] - \\ \text{KL}(Q(W,b) \parallel P(W,b)) + \lambda L(\theta) \end{aligned} \tag{15}$$

where the first term is Bayesian log-likelihood, which ensures accurate predictions, the Second term is KL-divergence, which acts as a Bayesian regularise, and the third term is PSO algorithm-based hyperparameter optimization optimizes the network for better generalisation. The state-of-the-art integration of the PSO algorithm with BLSTM NN is discussed, and its flowchart is given in the following section for a step-by-step understanding.

2.5. Flowchart of the proposed PSO-BLSTM NN model

The proposed framework integrates the PSO algorithm with a BLSTM NN to enhance forecasting accuracy for SPVP generation. The process is divided into two main sections: PSO-based hyperparameter optimization and BLSTM-based forecasting. In the initial phase, the PSO algorithm sets key parameters, including the number of particles, inertia weight, and cognitive and social coefficients [16]. Each particle represents a potential set of hyperparameters for the BLSTM model. The fitness of each particle is evaluated by training the BLSTM model and calculating the Root Mean Squared Error (RMSE). The best solutions are updated iteratively, guiding particles toward optimal hyperparameters. This process continues until convergence, ensuring the best set of hyperparameters is selected for forecasting.

The second phase involves training the BLSTM model using the optimized hyperparameters. Historical SPVP generation data is normalized before feeding it into the model for training. The BLSTM network, designed to capture both short-term and long-term dependencies, processes the data and generates forecasts. The process continues until a predefined performance criterion is met. By combining PSO for optimization with the predictive capabilities of BLSTM, the proposed model improves forecasting accuracy while minimizing error. The following section discusses experimental setups, including dataset details and computational platforms.

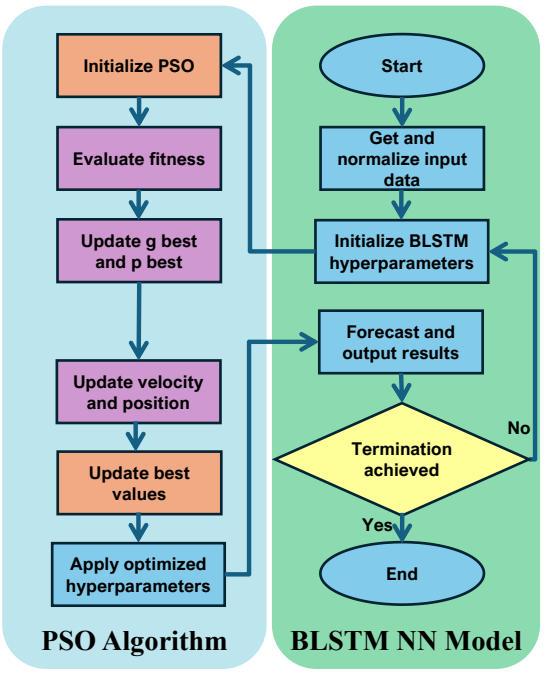

Fig. 2. Flowchart of the proposed PSO-BLSTM NN model.

3. Experimental Setup and Validation

This paper was meticulously designed to assess the effectiveness of a PSO algorithm-optimized BLSTM NN model for forecasting SPVP generation. The dataset includes five years of historical SPVP generation output and is split into training and testing, with 70% allocated for training and 30% for testing [24]. This distribution ensures that a significant portion of the data is used to train the model while enough remains for testing. The model's Bayesian framework allows for continuously updating the posterior distributions of these parameters. Meanwhile, the PSO algorithm optimizes the hyperparameters, striving for optimal model performance. The experimental setups with initial values are given in Table 2.

The validation includes various analyses through different metrics [25]. Assessing the percentage improvement from a standard LSTM NN model to the BLSTM NN model and further enhancement when applying the PSO algorithm. The validation metrics compare the performance across these models, emphasising accuracy and uncertainty estimation improvements. To benchmark the performance of the PSO algorithm-enhanced BLSTM NN model, several other models are employed for comparison. A basic LSTM NN model [26] without Bayesian inference or PSO algorithm serves as a baseline. The metrics used in the analyses are given below.

RMSE provides error units consistent with the forecasted values.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{16}$$

Mean Absolute Percentage Error (MAPE):

Table 2. The experimental setups for the PSO-BLSTM NN model.

Components	Details
System configuration	Intel core i7-12700K, 32GB RAM, NVIDIA RTX 3090 (24GB VRAM), 1TB NVMe SSD, Windows 11 Pro (OS)
Programming language	Python (Spyder) – TensorFlow 2.17
Optimization algorithm	PSO algorithm implemented using pySwarms
Dataset source	Real-world solar PV power data (Grid-connected farm)
Total data points	175,200
Feature set	Historical SPVP generation output
Training-test split	70% training, 30% testing
Data preprocessing	Min–Max scaling applied to input features
Neural network architecture	Two LSTM layers (64 and 32 units), Bayesian dense layer, one output neuron
Activation functions	ReLU (hidden layers), Linear (Output layer)
Loss function	Negative log-likelihood (NLL)
Solver type	RMSprop
PSO algorithm parameters	Swarm size = 30, Inertia weight (ω) = 0.7, Cognitive coefficient (C_1) = 1.8, Social coefficient (C_2) = 1.6, Maximum iterations = 150

$$\text{MAPE} = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{17}$$

R^2 indicates the proportion of variance in the dependent variable that is forecastable from the independent variables.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \tag{18}$$

Prediction Interval Coverage Probability (PICP):

$$\text{PICP} = \frac{1}{n}\sum_{i=1}^{n} 1(L_i \le y_i \le U_i) \tag{19}$$

Prediction Interval Normalized Average Width (PINAW):

$$\text{PINAW} = \frac{1}{n}\sum_{i=1}^{n}\frac{U_i - L_i}{\max(y) - \min(y)} \tag{20}$$

Table 3. Performance analysis.

Model	%RMSE	%MAPE	R^2	TT (hrs)
DB	23.83	39.02	0.72	4.53
RBF	21.42	36.77	0.75	5.26
Ensemble	19.89	35.82	0.77	8.16
RNN	16.76	33.56	0.79	8.31
LSTM NN model	13.49	28.23	0.84	10.22
BLSTM NN model	**11.44**	**22.51**	**0.89**	**10.45**
PSO-LSTM NN model	10.28	22.06	0.91	11.53
PSO-BLSTM NN model	**7.81**	**16.18**	**0.96**	**12.32**

Table 4. Performance improvement analysis.

Improvements	RMSE	MAPE	R^2	TT (hrs)
Due to the Bayesian integration	15.20%	20.26%	5.95%	-23 min
Due to the PSO optimization	31.73%	28.12%	7.87%	-1hr 47 min

where y_i is the actual SPVP value, $\hat{y}_i$ is the forecasted SPVP value, n is the total number of data points, $\bar{y}_i$ is the mean of actual values, L_i and U_i represent the lower and upper bounds of the prediction interval, respectively, and max (y) and min (y) are maximum and minimum values of y_i. The experimental setup and metrics used help highlight the proposed methodology's benefits and limitations, evaluate each model component, and offer valuable insights for future research in SPVP generation forecasting. In the following section, the performance of the models is analysed in various scenarios and discussed in detail to support long-term power grid planning.

Table 5. Uncertainty analysis.

Climatic condition	PICP (%)	PINAW
Clear day (05.04.2019)	97.2	0.121
Partly cloudy day (26.07.2019)	92.3	0.136
Cloudy day (11.09.2019)	90.4	0.143
Rainy day (02.11.2019)	87.7	0.158

Table 6. Scalability analysis.

Dataset size in years	$\%RMSE$	$\%MAPE$	TT (hrs)
2 years	16.33	29.68	6.26
3 years	13.43	27.36	8.17
4 years	10.76	23.21	10.55
5 years	**7.81**	**16.18**	**12.32**

4. Results and Discussion

This section presents the performance evaluation of the proposed PSO-BLSTM NN model using various analyses, such as accuracy, uncertainty, scalability, and sensitivity analyses conducted to validate the robustness and adaptability of the proposed model. The results are compared against benchmark models such as DB, RBF, ensemble methods, RNN, LSTM NN model, BLSTM NN model, and PSO-LSTM NN model.

4.1. Accuracy analysis

The accuracy analysis is carried out by comparing the performances of proposed and baseline models. The comparative analysis is presented in Table 3. The performance for different types of days is sampled from the forecasted output and shown in Fig. 3 to showcase the SPVP forecasted. The proposed PSO-BLSTM NN model outperforms the baseline models by achieving the lowest error values in % RMSE and % MAPE while attaining the highest R^2 scores in tested scenarios. The significant reduction in error shows the effectiveness of incorporating the PSO algorithm in optimizing the BLSTM NN model parameters. The improvement in accuracy is analysed to implement the Bayesian concept with the LSTM NN model and the PSO algorithm applied to the BLSTM NN model. The improvement results are given in Table 4. The accuracy analysis across different models highlights the impact of integrating techniques, such as Bayesian inference and the PSO algorithm. The methods enhanced forecasting accuracy and provided a more reliable assessment of forecast uncertainty.

Integrating the Bayesian approach into the LSTM NN model led to a 15.20% reduction in RMSE and a 20.26% reduction in MAPE. This underscores the BLSTM NN model's capability to provide more accurate and reliable forecasts by effectively capturing uncertainty. Further, to enhance model performance, the PSO algorithm achieves a 31.73% and 28.12% reduction in RMSE and MAPE, respectively.

4.2. Uncertainty analysis across different climatic conditions

Uncertainty analysis is vital for assessing the reliability of forecasting models under different weather conditions that influence SPVP generation. To validate the robustness of the PSO-BLSTM NN model, forecasting intervals are analysed under various climatic conditions, including clear days, partially cloudy days, cloudy days, and rainy-day scenarios.

The visualisation is shown in Fig. 3; the PICP and PINAW are the metrics used for the uncertainty analysis. The results obtained

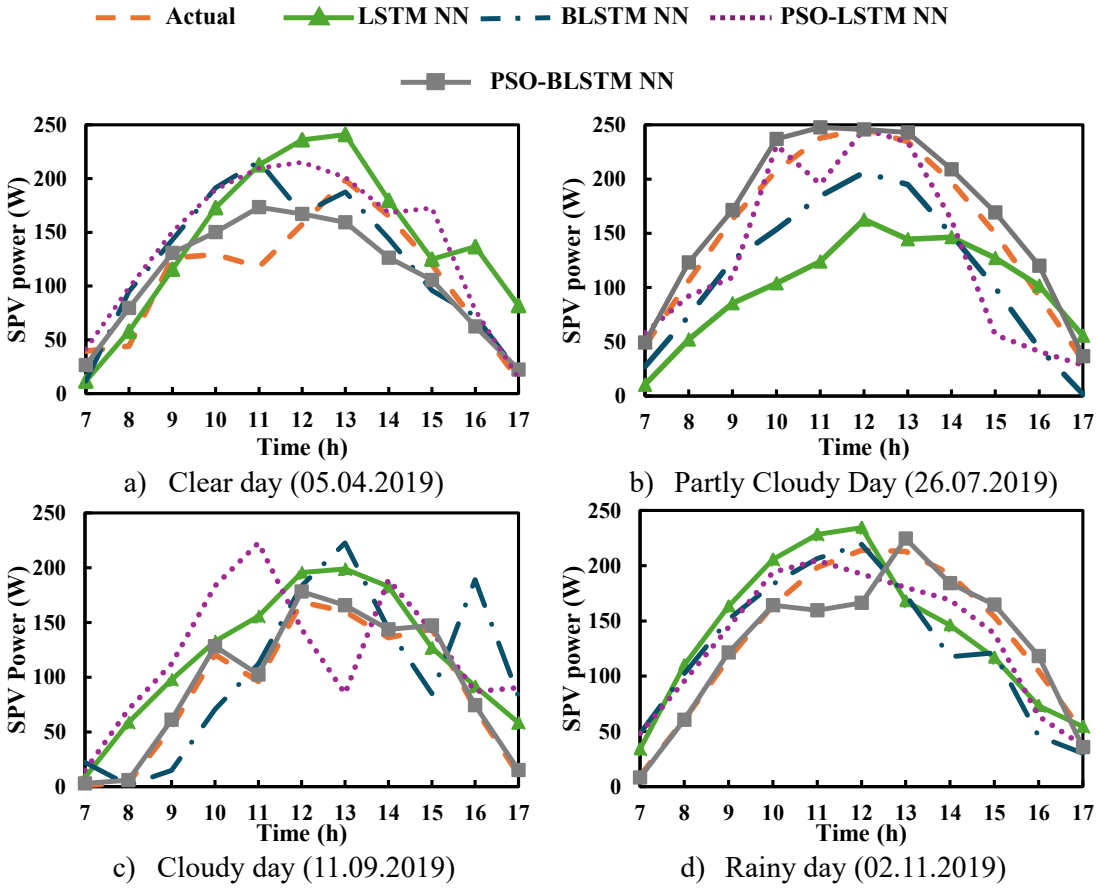

Fig. 3. Performance comparison of the proposed model in different types of days.

Table 7. Sensitivity analysis of PSO algorithm parameters.

Range of inertia weight (ω)	Range of cognitive coefficient (C_1)	Range of social coefficient (C_2)	Convergence speed	Experimental inference
0.9 – 0.4	1.0 – 2.0	1.0 – 2.0	Slow	Low accuracy, premature convergence
0.8 – 0.5	2.0 – 2.5	1.5 – 2.0	Fast	Overfitting observed
0.7 (constant)	1.5 – 1.6	2.0 – 2.5	Moderate	Stable, but slight accuracy drops
0.6 – 0.3	1.2 – 1.5	1.0 – 1.5	Slow	Low accuracy, poor convergence
0.9 – 0.4	**1.6 – 1.8**	**1.2 – 1.4**	**Balanced**	**High accuracy, stable performance**
0.5 (constant)	2.0 – 2.2	1.8 – 2.0	Unstable	Moderate accuracy, oscillations

Table 8. PSO algorithm parameters.

PSO parameters (ω, C_1, C_2)	$\%RMSE$	$\%MAPE$	TT (hrs)
(0.9, 1.8, 1.2)	7.81	16.18	12.32

are shown in Table 5. The model performs exceptionally well under stable sunny conditions, achieving the highest PICP (97.2%) and the lowest PINAW (0.121), indicating high confidence in forecasting with minimal uncertainty. There is a slight decrease in forecasting reliability under cloudy weather, with PICP dropping to 95.4% and PINAW increasing to 0.133, reflecting the increased variability in solar power generation. The highest level of uncertainty is observed under rainy conditions, where PICP reduces to 92.7% and PINAW increases to 0.148. This is expected, as heavy cloud cover and unforecastable weather significantly affect solar power generation. These results confirm that while the proposed PSO-BLSTM NN model maintains reliable forecasting intervals across various climatic conditions, the uncertainty increases as weather conditions become more unforecastable. This insight is valuable for energy management systems that must factor in climatic variability in forecasting SPVP generation output.

4.3. Scalability analysis

The adaptability of the proposed model is tested for large-scale applications. The performance of the model is evaluated across datasets of different sizes. The scalability analysis results are given in Table 6. It resembles that the PSO-BLSTM NN model maintains reasonable percentages of RMSE and MAPE for all sizes of datasets. In the case of the training time, it is inversely proportional to the data size, as expected in any deep learning technique. The scalability of the proposed model is ensured through the analysis report with minimum deviations in results for different scales of datasets.

4.4. Sensitivity analysis of PSO algorithm parameters

The sensitivity of the proposed model to PSO algorithm control parameters is analysed by varying inertia weight (ω), cognitive coefficient (C_1), and social coefficient (C_2) through the investigation in the literature and trial and error experimentation method. Table 7 presents the sample of the test report to understand the impact of the parameters on model performances, and Table 8 gives the final extracted values from the test report for the model operation. The results indicate that an optimal balance between C_1 and C_2 ensures better convergence and improved accuracy. A very high C_1 leads to overfitting, while an excessively high C_2 reduces generalisation capability. In such a way, the values for the parameters bolded in row 5 of Table 7 are used to train and test the proposed forecasting model.

5. CONCLUSION AND FUTURE SCOPE

The paper introduces a BLSTM NN model optimized with the PSO algorithm to improve the accuracy of SPVP generation forecasting. The model is trained using five years of historical data from a 1 MW solar power plant, recorded at 15-minute intervals. The proposed model is evaluated against baseline models to assess its forecasting accuracy, ability to quantify uncertainty, scalability, and sensitivity. The results show that the PSO-BLSTM model outperforms baseline models by reducing the RMSE by 16.02%, lowering the MAPE by 22.84%, and improving the R^2 score by 24% compared to the conventional baseline DB model. Integrating Bayesian inference further strengthens the ability of the model to quantify forecasting uncertainty, making it a reliable model for power grid planning. The proposed model's future scope also focuses on applications in other time-series forecasting domains, such as finance, healthcare, and climate modeling. The work provides valuable insights for long-term applications, and further research can be extended to improve computational efficiency and reduce training time for the rapid acceptance of short-term forecasting applications.

REFERENCES

[1] M. H. Mousavi, H. M. CheshmehBeigi, and M. Ahmadi, "A ddsrf-based vsg control scheme in islanded microgrid under unbalanced load conditions," *Electr. Eng.*, vol. 105, no. 6, pp. 4321–4337, 2023.

[2] J. V. P.R., N. S. K., V. T., J. G. A., and V. Arunachalam, "Development of a long-term solar pv power forecasting model for power system planning," *World J. Eng.*, 2024.

[3] 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, p. 109792, 2020.

[4] R. A. Rajagukguk, R. A. A. Ramadhan, and H. J. Lee, "A review on deep learning models for forecasting time series data of solar irradiance and photovoltaic power," *Energies*, vol. 13, no. 24, 2020.

[5] H. Sharadga, S. Hajimirza, and R. S. Balog, "Time series forecasting of solar power generation for large-scale photovoltaic plants," *Renew. Energy*, 2020.

[6] S. Frizzo Stefenon, C. Kasburg, A. Nied, A. C. Rodrigues Klaar, F. C. Silva Ferreira, and N. Waldrigues Branco, "Hybrid deep learning for power generation forecasting in active solar trackers," *IET Gener. Transm. Distrib.*, vol. 14, no. 23, pp. 5667–5674, 2020.

[7] M. N. Akhter, S. Mekhilef, H. Mokhlis, Z. M. Almohaimeed, M. A. Muhammad, A. S. M. Khairuddin, R. Akram, and M. M. Hussain, "An hour-ahead pv power forecasting method based on an rnn-lstm model for three different pv plants," *Energies*, vol. 15, no. 6, p. 2243, 2022.

[8] S. M. J. Jalali, S. Ahmadian, A. Kavousi-Fard, A. Khosravi, and S. Nahavandi, "Automated deep cnn-lstm architecture design for solar irradiance forecasting," *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 52, no. 1, pp. 54–65, 2022.

[9] L. Yang, M. Xu, Y. Guo, X. Deng, F. Gao, and Z. Guan, "Hierarchical bayesian lstm for head trajectory prediction on omnidirectional images," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021.

[10] J. Wu, X. Chen, H. Zhang, L.-D. Xiong, H. Lei, and S. Deng, "Hyperparameter optimization for machine learning models based on bayesian optimization," *J. Electron. Sci. Technol.*, 2019.

[11] J. Chen, D. Pi, W. Zhiyuan, X. Zhao, Y. Pan, and Q. Zhang, "Imbalanced satellite telemetry data anomaly detection model based on bayesian lstm," *Acta Astronaut.*, 2021.

[12] D. Wu, Z. Jiang, X. Xie, X. Wei, W. Yu, and R. Li, "Lstm learning with bayesian and gaussian processing for anomaly detection in industrial iot," *IEEE Trans. Ind. Inform.*, 2020.

[13] Y. Guo, J.-Y. Li, and Z.-H. Zhan, "Efficient hyperparameter optimization for convolution neural networks in deep learning: A distributed particle swarm optimization approach," *J. Stat. Comput. Simul.*, 2021.

[14] K. J. Iheanetu, "Solar photovoltaic power forecasting: A review," *Sustainability*, vol. 14, no. 24, p. 17005, 2022.

[15] O. Koduri, R. Ramachandran, and M. Saiveerraju, "Empirical mode decomposition and optimization assisted ann based fault classification schemes for series capacitor compensated transmission line," *J. Oper. Autom. Power Eng.*, vol. 13, no. 1, pp. 52–73, 2025.

[16] W. Zhang, D. Ma, J. J. Wei, and H. F. Liang, "A parameter selection strategy for particle swarm optimization based on particle positions," *Expert Syst. Appl.*, 2014.

[17] M. F. Rezaei, M. Gandomkar, and J. Nikoukar, "Optimizing multi-objective function for user-defined characteristics relays and size of fault current limiters in radial networks with renewable energy sources," *J. Oper. Autom. Power Eng.*, vol. 12, no. 1, pp. 42–53, 2024.

[18] D. Mora-Mariano and A. Flores-Tlacuahuac, "Bayesian lstm framework for the surrogate modeling of process engineering systems," *Comput. Chem. Eng.*, 2024.

[19] U. Kubayev, S. Toshalieva, I. Ayubov, M. Farxodjon, Q. F. Ergash Ugli, Z. Jakhongir Rasulovich, T. B. Nietbaevich, A. A. Bektemirov, A. S. Seyranovna, R. Haydarovich Kushatov *et al.*, "Adaptive islanding detection in microgrids using deep learning and fuzzy logic for enhanced stability and accuracy," *J. Oper. Autom. Power Eng.*, vol. 12, no. Special Issue, pp. 33–42, 2024.

[20] X. Ren, S. Liu, Y. Xiaodong, and D. Xia, "A method for state-of-charge estimation of lithium-ion batteries based on pso-lstm," *Energy*, 2021.

[21] C. Zhang, J. Butepage, H. Kjellstrom, and S. Mandt, "Advances in variational inference," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 2008–2026, 2019.

[22] G. Tiwari and S. Saini, "Optimizing fault identification in power distribution systems by the combination of svm and deep learning models," *J. Oper. Autom. Power Eng.*, 2024.

[23] Y.-C. Liu and P.-D. Leifsson, "Analysis of agricultural and engineering systems using simulation decomposition," in *Comput. Sci. – ICCS 2022*. Cham: Springer, 2022, pp. 435–444.

[24] D. Talgatkyzy, N. Haroon, S. Hussein, S. K. Ibrahim, K. Jabbar, B. Mohammed, and S. Hameed, "Renewable energy resources development effect on electricity price: an application of machine learning model," *J. Oper. Autom. Power Eng.*, vol. 11, no. Special Issue, 2023.

[25] S. J. S. Kumar and K. N. Sam, "Multi-hybrid stl-lstm-sde-ma model optimized with iwoa for solar pv-power forecasting," in *Proc. IEEE Int. Conf. Power Electron. Smart Grid Renew. Energy*, 2023, pp. 1–6.

[26] N. Almuratova, M. Mustafin, K. Gali, M. Zharkymbekova, D. Chnybayeva, and M. Sakitzhanov, "Enhancing microgrid resilience with lstm and fuzzy logic for predictive maintenance," *J. Oper. Autom. Power Eng.*, vol. 12, no. Special Issue (Open), pp. 7–15, 2024.