Computers, Materials & Continua
DOI:10.32604/cmc.2022.021447
Article

Tech Science Press

Big Data Analytics Using Swarm-Based Long Short-Term Memory for Temperature Forecasting

Malini M. Patil[1,*], P. M. Rekha[1], Arun Solanki[2], Anand Nayyar[3,4] and Basit Qureshi[5]

[1]JSS Academy of Technical Education, Bengaluru, 560060, India
[2]Gautam Buddha University, Greater Noida, 201312, India
[3]Graduate School, Duy Tan University, Da Nang, 550000, Vietnam
[4]Faculty of Information Technology, Duy Tan University, Da Nang, 550000, Vietnam
[5]College of Computer & Info Sc, Prince Sultan University, Saudi Arabia
[*]Corresponding Author: Malini M. Patil. Email: malinimpatil@jssateb.ac.in

Received: 03 July 2021; Accepted: 27 September 2021

Abstract: In the past few decades, climatic changes led by environmental pollution, the emittance of greenhouse gases, and the emergence of brown energy utilization have led to global warming. Global warming increases the Earth's temperature, thereby causing severe effects on human and environmental conditions and threatening the livelihoods of millions of people. Global warming issues are the increase in global temperatures that lead to heat strokes and high-temperature-related diseases during the summer, causing the untimely death of thousands of people. To forecast weather conditions, researchers have utilized machine learning algorithms, such as autoregressive integrated moving average, ensemble learning, and long short-term memory network. These techniques have been widely used for the prediction of temperature. In this paper, we present a swarm-based approach called Cauchy particle swarm optimization (CPSO) to find the hyperparameters of the long short-term memory (LSTM) network. The hyperparameters were determined by minimizing the LSTM validation mean square error rate. The optimized hyperparameters of the LSTM were used to forecast the temperature of Chennai City. The proposed CPSO-LSTM model was tested on the openly available 25-year Chennai temperature dataset. The experimental evaluation on MATLABR2020a analyzed the root mean square error rate and mean absolute error to evaluate the forecasted output. The proposed CPSO-LSTM outperforms the traditional LSTM algorithm by reducing its computational time to 25 min under 200 epochs and 150 hidden neurons during training. The proposed hyperparameter-based LSTM can predict the temperature accurately by having a root mean square error (RMSE) value of 0.250 compared with the traditional LSTM of 0.35 RMSE.

Keywords: Climatic change; big data; temperature; forecasting swarm intelligence; deep learning

1 Introduction

In today's world, climatic change is a global issue for all countries. In India, climatic change has a significant effect on agriculture. The term climatic change refers to the variation in seasonal behavior of the climate. It occurs by increasing the temperature artificially by burning fuels, oils, and gases, leading to the exhalation of greenhouse gases. The climatic change causes the following effects, such as variation in temperature, rainfall, and ocean temperature, and storms. It also shrinks the glaciers and changes the seasonal pattern for agriculture. Among these many effects, the temperature variation causes vast problems to human beings and agriculture. Hence, preventive measures and learning are required to reduce climatic change and predict climatic changes on the basis of past data. An optimized machine learning approach is proposed in this paper to forecast the temperature by using big data. This section discusses the techniques used in predicting climatic change. Fuzzy logic can predict the hotness value for a short period only. Temperature forecasting is used to predict the atmosphere in the future and has wide application in several areas, such as gauging air flight delay, satellite dispatching, crop creation, and natural calamities. Although neural networks are the mainstream of machine learning methods, they have not gained prominence in the last part due to the inclination plummet issue and the capacity of personal computers prompting extremely high learning time. With the advancement of innovation, different strategies, such as equal handling, high-performance computing groups, and graphics processing units, have been created. These strategies improve their presentation when applied in neural networks.

The mean temperature and precipitation value are used to forecast the weather. On the basis of the forecasted results, the lagged data with lesser time intervals can predict the condition better compared with the tuned network. The performance of machine learning and traditional approaches is equal for the smaller dataset. These approaches utilize 75-year data to predict the 15-year weather condition with a minimum error rate. The long short-term memory (LSTM) network uses the reduced sample points in forecasting. Among the three methods, the recurrent neural network can predict the weather accurately due to the five-layered architecture and linearization process of data. The combination of M-ARIMA and neural network outperforms the k-nearest neighbor (KNN) prediction results. The results indicate that the machine learning approaches can predict the weather accurately at 2 m humidity conditions, which is greater than the traditional weather forecasting model by IUM. The daily data are split through Fourier transform to analyze their features effectively. The extracted features are reduced through principal component analysis, and Elman-based backpropagation network is used for temperature forecasting. This approach aims to reduce the computation time and improve the result of root mean square error (RMSE) and mean absolute error (MEA).

The rest of this paper is organized as follows: Section 2 discusses the literature review of the techniques used for weather forecasting. Section 3 introduces the conventional approaches in finding the hyperparameters of LSTM. Section 4 explains the proposed system model Cauchy particle swarm optimization (CPSO)-LSTM briefly. Section 5 presents the experimental setup, results, and discussions. Section 6 provides the conclusions with a summary of the CPSO-LSTM technique in temperature forecasting using big data.

2 Literature Survey

Chen et al. [1] proposed a fuzzy-based approach to forecast the temperature in Taipei. A sliding window approach is used to predict the hotness value for short intervals by using a smaller dataset. An ensemble approach was used in Maqsood et al. [2] to estimate the temperature in

Canada. This approach can process large datasets compared with fuzzy logic. Shrivastava et al. [3] surveyed different machine learning approaches in forecasting the weather conditions in a region. These techniques require more time for learning the data during prediction. Sawaitul et al. [4] and Naik et al. [5] utilized a backpropagation algorithm to predict weather conditions. The learning parameters of the traditional Levenberg–Marquardt algorithm are modified; Shereef et al. [6] used these learning parameters to forecast the weather conditions in Chennai on the basis of 2010 data.

Papacharalampous et al. [7,8] combined different time series models on a four-year weather data to forecast the one-month weather condition. Random walks and a naive algorithm are utilized to split the four-year data. Trends, autoregressive integrated moving average (ARIMA), and prophet techniques are used to forecast the weather. The seasonal trend with the prophet technique can accurately predict the weather compared with the traditional approaches. Yahya et al. [9] combined fuzzy clustering, ARIMA, and artificial neural network to forecast the weather conditions for two decades in an Iraqi province.

Shivhare et al. [10] proposed a time series model called ARIMA to forecast the weather conditions, such as temperature and rainfall, in Varanasi. Graf et al. [11] improved the performance of the artificial neural network by using wavelet transform to predict the river temperature. Here, wavelet transform is applied to decompose the averaged time-series data of water and air temperature. The decomposed data are used to forecast the water temperature by using a multilayer perceptron neural network. Singh et al. [12] investigated machine learning approaches, such as support vector machines, artificial neural networks, and recurrent neural networks in predicting the temperature. The time series model (ARIMA) is improved by using the map-reduce algorithm to adjust the temperature sample points. Bendre et al. [13] trained the obtained sample points from M-ARIMA with the KNN and neural network to predict the temperature. The Institute of Urban Meteorology, China predicted the surface air temperature by using artificial intelligence techniques. The study on this is presented by Lan et al. [14]. Ji et al. [15] conducted temperature forecasting by using frequency transformation and a backpropagation network. Attoue et al. [16] used a simple feedforward neural network approach to forecast the indoor temperature. Zhang et al. [17] used a hybrid combination of machine learning techniques to forecast the temperature in southern China. Ensemble empirical mode decomposition is utilized to identify the sample points for network training by using partial autocorrelation function.

Spencer et al. [18] proposed a refinement process for tuning the lasso regression parameter to reduce the data in forecasting the indoor temperature. Accurate prediction is achieved only through the perfect learning of information. On the basis of this learning mechanism, Fente et al. [19] used a deep recurrent neural network to forecast the weather by using different environmental factors. The authors investigated the importance of machine learning's role in weather prediction. All categories of machine learning algorithms, such as supervised, unsupervised, and clustering approaches, are analyzed. The literature review shows that the survey analysis conducted by Spencer et al. and Fente et al. helps to improve the algorithms and their forecasting results, as presented by Kunjumon et al. [20]. Similarly, Mahdavinejad et al. [21] explored the importance machine learning in the Internet of Things (IoT) data prediction.

Wang et al. [22] proposed a deep learning technique with a different validation parameter to predict future weather conditions. Sun et al. [23], Jane et al. [24], and Sharma [25] investigated the importance of machine learning in forecasting environmental conditions. All machine learning algorithms, such as supervised, unsupervised, and clustering methods, are analyzed. Tripathy et al. [26] examined the importance of machine learning in big data. All machine learning algorithms, such as supervised, unsupervised, and clustering approaches, are analyzed. Qiu et al. [27]

predicted the river temperature by using a long short-term neural network. The temporal effects of the temperature in the river area are analyzed on the basis of the forecasted values by LSTM. Lee et al. [28] explored different machine learning algorithms like, such as multilayer perceptron, convolution neural network, and recurrent neural network, in forecasting the daily temperature.

Ozbek et al. [29] evaluated different time interval temperature variations for atmospheric air by using a recurrent neural network. The authors in [30] used a LSTM network to predict the indoor air temperature by using IoT data. Sorkun et al. [31] and Maktala et al. [32] introduced a slight modification in LSTM operations to forecast the temperature variations Barzegar et al. [33] proposed a water quality monitoring system with two quality variables, namely, dissolved oxygen and chlorophyll-a, by using LSTM and convolutional neural network (CNN) models. They built a coupled CNN-LSTM model, which is extremely efficient. Vallathan et al. [34] presented an IOT-based deep learning model to predict the abnormal events occurring in daycare and crèches to protect them from abuses. The footages of networked surveillance systems are used as input. The main goal in the paper published by Movassagh et al. [35] is to train the neural network by using metaheuristic approaches and to enhance the perceptron neural network precision. Ant colony and invasive weed optimization algorithms are used for performance evaluation. Artificial neural networks use various metaheuristic algorithms, including approximation methods, for training the neural networks.

3 Conventional Approaches

In the existing approach, the network hyperparameters are determined through trial and error and grid search method. These methods consume more computational time for selecting the parameters. They are also an exhaustive search method with minimal output efficiency. An optimized approach (CPSO) is proposed to find the LSTM network hyperparameters and to overcome this problem. The traditional LSTM algorithm is used to forecast the temperature. The hyperparameter tuning is performed by using grid search, as shown in Fig. 1, and the steps are as follows.

(1) The input dataset is split into training and testing data by using a hold-out approach.
(2) The hyperparameter for the recurrent neural network is initialized.
(3) Train the network and check the RMSE value.
(4) If Val RMSE<0.4
(5) The hyperparameters are obtained.
(6) Else
(7) Change the value of the network.
(8) Repeat step 3.
(9) End if.
(10) Train and test the neural network with the tuned parameters and data.

This approach can determine the temperature effectively, but it consumes high processing time for computing the LSTM hyperparameters. Hence, this drawback is overcome by using the proposed optimized approach-based LSTM.

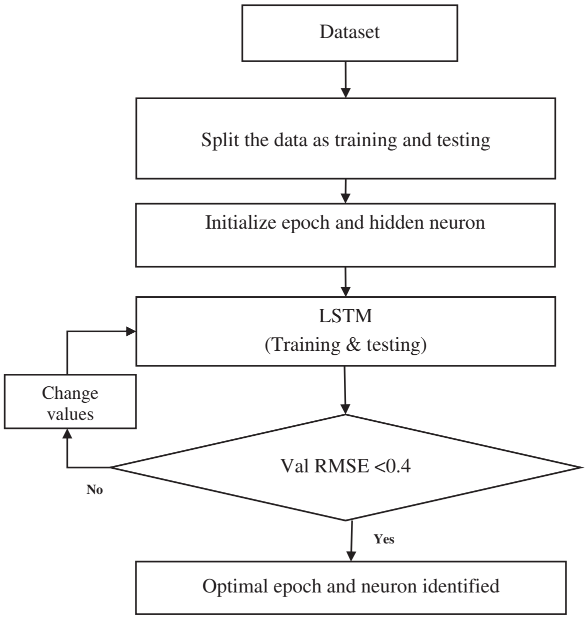

Dataset
Split the data as training and testing
Initialize epoch and hidden neuron
LSTM
(Training & testing)
Change values
Val RMSE <0.4
No
Yes
Optimal epoch and neuron identified

Figure 1: Temperature forecasting using grid search-based LSTM

4 System Model

This section explains the optimization of a LSTM network for temperature prediction by using the proposed method. The proposed method is tested on the Chennai temperature dataset with two decades of information.

4.1 Dataset

The average daily temperatures posted on this site are obtained from the Global Summary of the Day (GSOD) dataset and are computed from 24 h temperature readings. Some previous datasets compiled by the National Climatic Data Center, such as the local climatological data monthly summary, contained daily minimum and maximum temperatures but did not contain the average daily temperature computed from 24 h readings. The temperature dataset is downloaded from the website [36] for the Chennai region. The dataset comprises four columns. The first column is the month, the second column is the date, the third column is the year, and the last column is the temperature readings in Fahrenheit. The data are collected from January 1995 to May 2020. Therefore, the total data comprise 9,266 temperature records. This information is used to forecast the temperature by using the proposed CPSO-LSTM network. The average daily temperatures posted on this site are computed from 24 h temperature readings in the GSOD data.

4.2 Preprocessing

In this part, the input data are checked to find the missing records. The unavailable data are replaced with the average temperature of that month. The preprocessed data are split into training and testing data by using the hold-out method. The hold out ratio is 0.7%, and 70% of data is the training data and 30% as testing data. The hold-out approach is used to split the data due to the abundant temperature records. The hold-out approach uses the ratio technique to divide the data. The data with a more significant number of records are the training data and the remaining as testing data, as shown in Fig. 2.

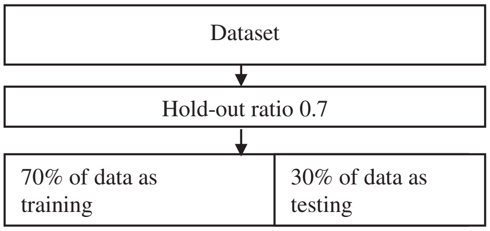


Figure 2: Splitting the data as training and testing

4.3 Hyperparameter Tuning Using Particle Swarm Optimization

The training data from the preprocessed dataset are used to determine the hyperparameters of the deep learning network (epoch and number of hidden neurons) by using CPSO. CPSO uses a single objective function to determine the LSTM network tuning parameters. Eq. (1) shows the objective function for CPSO.

$$PSO_{OF} = (Error\, rate_{Temp}). \tag{1}$$

The term error rate indicates the difference between the actual and predicted temperatures by using the LSTM network, and its formula is given in Eq. (2).

$$Error\, rate = Actual\ temperature - Predicted\ temperature_{LSTM}. \tag{2}$$

Particle swarm optimization is inspired by the bird's nature to find the optimal landing position for its group (i.e., a flock of birds). Each bird moves in a random direction to determine the safe place for landing. On the basis of this landing analysis of birds, particle swarm optimization is proposed in [37].

Here, the landing position is considered the objective function, and the birds are considered the particles. The proper solution is achieved by defining the search region; otherwise, the algorithm performs its operation. The search space for CPSO is defined by mentioning the upper and lower bounds of the searching variables. The particles are located randomly with different solutions. A particle moves with a velocity within its search region and updates its position and velocity for each iteration. In each iteration, the results are updated for each particle and its minimum value as local best. At the end of iterations, the overall minimum value is stored as the global best solution for the problem.

The LSTM network tuning parameters are determined by solving Eq. (1) using the N number of particles in the CPSO algorithm. These N particles have two dimensions. One denotes the epoch, and the other denotes the hidden neurons. The position of the particles is denoted as P_{N1} and P_{N2}, where 1 and 2 indicate the dimensions and is commonly indicated as P_{Nd}. The particle position update equation for each iteration is given in Eq. (3).

$$P_{ND}(I) = (P_j(I)) + (P_j(I)) - P_{i,j}(I), \tag{3}$$

$$P_{Nd}(I+1) = P_{Nd}(I) + V_{Nd}(I+1), \tag{4}$$

where V_{Nd} is the velocity for moving a particle from one place to another. The particle's velocity is updated for each iteration by using Eq. (4).

$$V_{Nd}(I+1) = \omega * V_{Nd}(I) + LC_1 * rand1 * (L_{Nd} - P_{Nd}(I)) + LC_2 * rand2 * (G_{Nd} - P_{Nd}(I)), \tag{5}$$

$$G_{Nd}(I) = G_{Nd}(I) + CW(I) * R(minimum, maximum), \tag{6}$$

$$CW(I) = \frac{\sum_{j=1}^{N} V_{Nd}[j][I]}{N}. \tag{7}$$

The particle updates its velocity on the basis of its local best solution L_{Nd} and global best solution G_{Nd} by using Cauchy weight. The other terms are constant values, as shown in Tab. 1.

Table 1: Parameters for CPSO

Parameters	Values
Population (N)	50
Iterations (I)	100
Inertia weight (ω)	1.0
Learning coefficient (LC1)	1.5
Learning coefficient (LC2)	2.0
Rand 1 and Rand 2	Random value between 0 and 1
Lower bound	[100–50]
Upper bound	[300–250]

The pseudocode for the CPSO algorithm is designed by using Eqs. (1) to (7) and the parameters in Tab. 1, as shown in Fig. 3.

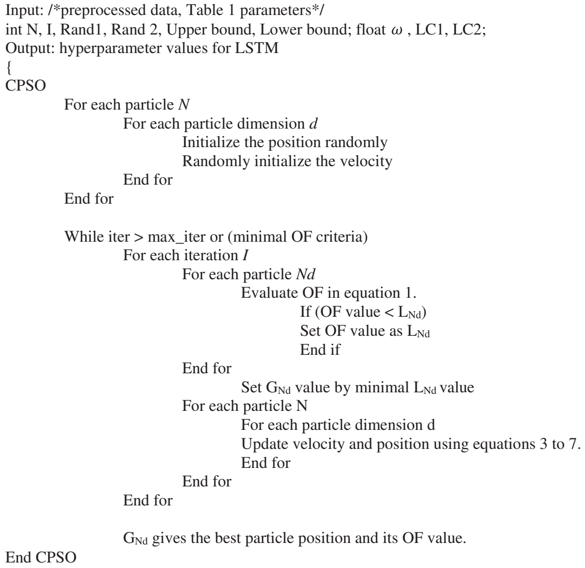
The optimal epochs and hidden neurons for temperature forecasting are determined for each particle. This value is used for the LSTM network. The pictorial representation of the proposed method is shown in Fig. 4.

The steps in the Cauchy particle swarm-based long short-term neural network for temperature forecasting are as follows.

(1) The input dataset is preprocessed to replace the missing information with its average value.
(2) The preprocessed data are split into training and testing data by using a hold-out approach.
(3) The hyperparameter for the recurrent neural network is determined from the CPSO by minimizing the mean square error rate.
(4) The neural network is trained and tested with the tuned parameters and data.
(5) The predicted output is evaluated in terms of the RMSE and MEA.

4.4 Temperature Forecasting Using Tuned Long Short-Term Neural Network

A long short-term neural network is a type of recurrent neural network that which repeats the neural network in chain structure, as shown in Fig. 5.



```
Input: /*preprocessed data, Table 1 parameters*/
int N, I, Rand1, Rand 2, Upper bound, Lower bound; float $\omega$ , LC1, LC2;
Output: hyperparameter values for LSTM
{
CPSO
        For each particle $N$
                For each particle dimension $d$
                        Initialize the position randomly
                        Randomly initialize the velocity
                End for
        End for

        While iter > max_iter or (minimal OF criteria)
                For each iteration $I$
                        For each particle $Nd$
                                Evaluate OF in equation 1.
                                        If (OF value < $L_{Nd}$)
                                        Set OF value as $L_{Nd}$
                                        End if
                        End for
                                Set $G_{Nd}$ value by minimal $L_{Nd}$ value
                        For each particle N
                                For each particle dimension d
                                Update velocity and position using equations 3 to 7.
                                End for
                        End for
                End for

                $G_{Nd}$ gives the best particle position and its OF value.
End CPSO
```

Figure 3: Pseudocode for the CPSO algorithm

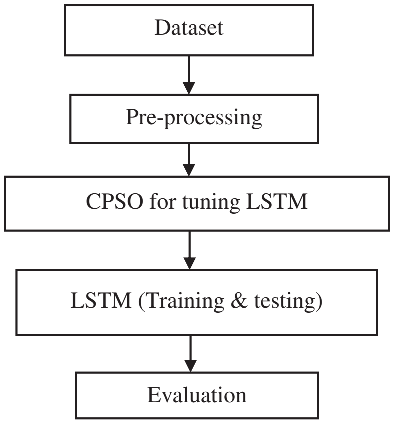


Figure 4: Temperature forecasting using CPSO-LSTM

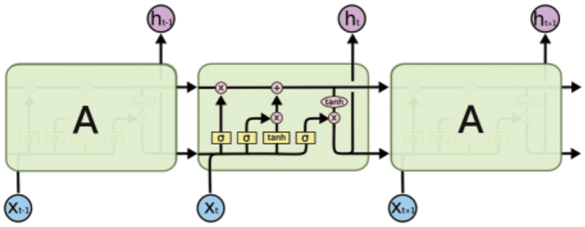


Figure 5: LSTM network

The repeating module for LSTM differs from the traditional neural network by having repeated neural networks (tanh and sigma) in the module. The pink circles indicate the pointwise operations performed on the data. The split arrow lines at the end of the module represent the data copy. The merged line at the start of the module indicates the concatenated value. The dark line signifies the vector transfer operations as per the work presented by Sherstinsky [38]. LSTMs process the temperature data as cell states and runs between the two modules. A gated structure is used; this structure can add or remove the data by using a sigmoid layer. This gated structure allows the data if they have a value of 1. Otherwise, the data are not allowed to pass through. Three sigmoid layers are used to protect the cell state of nature. The input for the cell state is the current value and previous module output. This value is first fed into the first sigmoid layer. This sigmoid layer forgets the previous state by setting its value to 0 and updates the state with the new values shown in Fig. 6.

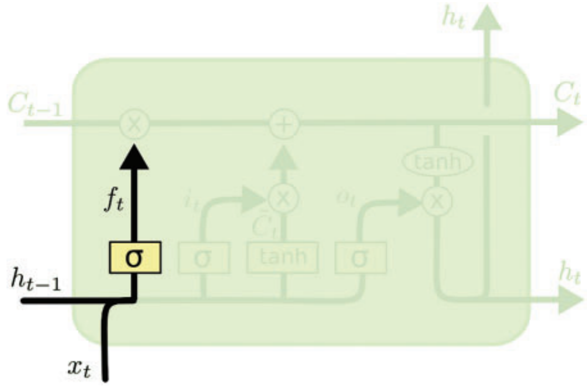


Figure 6: Forget operation using Sigmoid layer1

The output of the sigmoid layer is given in Eq. (8).

$$f_t = \sigma * (W_1[h_{t-1}, x_t] + wt_1) , \tag{8}$$

where W and wt are the weights of the sigmoid layer. The current state is added to the network by using the tanh function, as shown in Fig. 7.

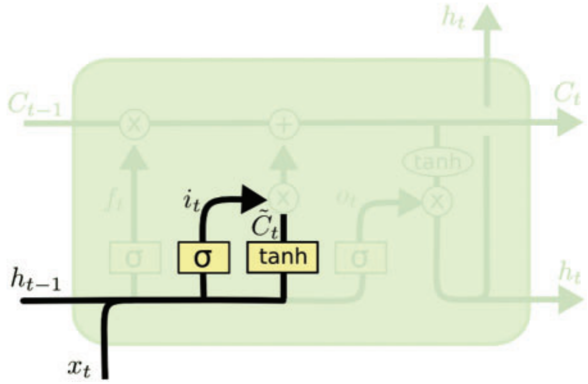


Figure 7: Replacing data using Sigmoid layer 2

The functions for replacing the data are given in Eqs. (9) and (10). The value is adjusted by using the tanh function to protect the cell state.

$$i_t = \sigma * (W_2[h_{t-1}, x_t] + wt_2) , \tag{9}$$

$$\hat{C}_t = \sigma * (W_{tan}[h_{t-1}, x_t] + wt_{tan}. \tag{10}$$

The previous cell state value is replaced with the current value by using Eqs. (9) and (10), as shown in Fig. 8. The replacement of the new value within the state is given in Eq. (11), and the current state is shown in Fig. 9.

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t. \tag{11}$$

This layer controls the operation of the cell state by updating the final cell state value, as shown in Eqs. (12) and (13).

$$o_t = \sigma * (W_3[h_{t-1}, x_t] + wt_3), \tag{12}$$

$$h_t = o_t * \ tanh\, tanh(C_t). \tag{13}$$

The above operations are repeated for the 150 hidden neurons to forecast the temperature. From the above process, the cell state can update its new value by using the final sigmoid layer. The LSTM performs this operation for 200 epochs to train the data. The testing process is conducted with the trained network. The LSTM forecasting efficiency is estimated by finding the difference between the predicted and actual temperatures.

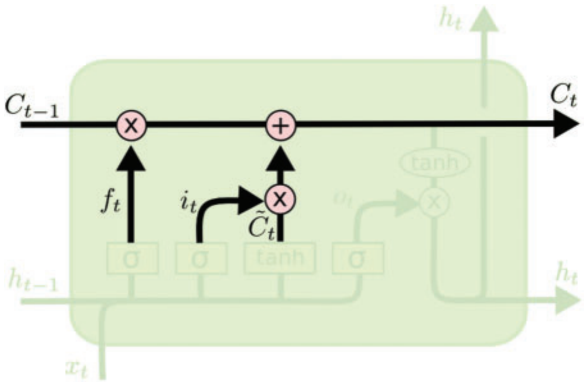


Figure 8: Updating the values

4.5 Evaluation

The forecasted output is evaluated by calculating the RMSE and mean absolute error (MAE) values. The formulas for the two parameters are given in Eqs. (14) and (15). The RMSE and MAE values should be the minimum for better forecasting results.

$$\text{RMSE} = \sqrt{\sum_{i=1}^{N} \frac{(Temp_{actual(i)} - Temp_{predict(i)})}{total\, records\, (N)}}, \tag{14}$$

$$\text{MAE} = \sum_{i=1}^{N} \frac{(Temp_{actual(i)} - Temp_{predict(i)})}{total\, records\, (N)}. \tag{15}$$

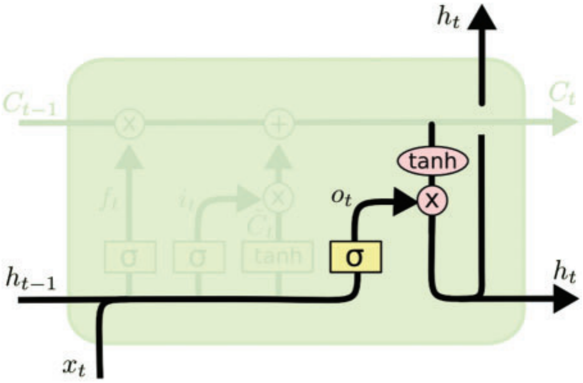
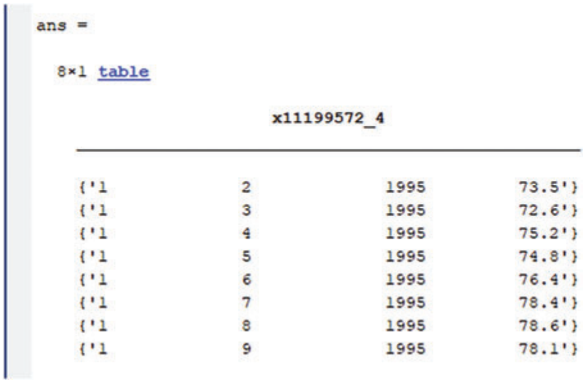


Figure 9: Cell state update

5 Experiments and Discussions

The proposed CPSO-LSTM for temperature forecasting is implemented on MATLAB R2020b version under Windows 10 environment. The sample data for the forecast process are given in Fig. 10.



```
ans =

  8×1 table

                 x11199572_4
    ______________________________________

    {'1        2          1995        73.5'}
    {'1        3          1995        72.6'}
    {'1        4          1995        75.2'}
    {'1        5          1995        74.8'}
    {'1        6          1995        76.4'}
    {'1        7          1995        78.4'}
    {'1        8          1995        78.6'}
    {'1        9          1995        78.1'}
```

Figure 10: Sample screenshot of data

The CPSO algorithm utilizes the preprocessed data to find the LSTM network tuning parameters. The CPSO algorithm determines the optimal parameter after 30 iterations. Fig. 11 shows the convergence curve for evaluating the fitness function.

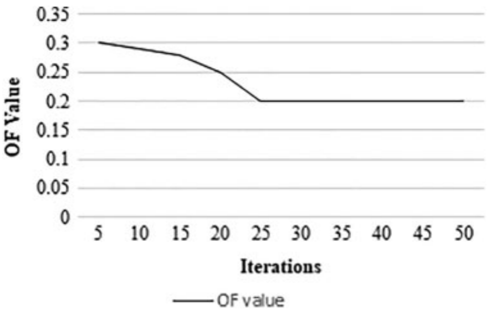


Figure 11: CPSO-LSTM results

The optimal epoch for the LSTM network is 200, and the optimal hidden neuron number is 150 for forecasting the temperature. With this optimal parameter, the LSTM network is trained and tested by using the preprocessed data. The estimated temperature is used to evaluate the evaluation metrics. The values are tabulated in Tab. 2.

Table 2: CPSO-LSTM evaluation

Parameters	Values
MAE	0.0625
RMSE	0.250

The hyperparameter comparison between the optimized and grid-search method LSTM network is tabulated in Tab. 3.

Table 3: CPSO-LSTM *vs.* LSTM evaluation

Parameters	Grid search-LSTM	CPSO-LSTM
Epoch	250	200
Hidden neuron	200	150
Computational time (min)	30	15

The proposed particle swarm optimization reduces the computational time for calculating the network hyperparameter. This condition is achieved by using more search agents to find the optimal solution. In grid search, the values are initialized randomly by the user, leading to a high processing time and a higher number of hyperparameter values. Tab. 4 shows that CPSO can reduce computational time by finding optimal network hyperparameters. The proposed method performance is compared with the traditional LSTM in terms of RMSE, MAE, and computational time.

Table 4: CPSO-LSTM *vs.* LSTM evaluation

Parameters	LSTM	CPSO-LSTM
MAE	0.1225	0.0625
RMSE	0.35	0.250
Computational time (min)	45	25

The comparisons in Tab. 4 depict that the proposed method requires less time compared with the traditional LSTM. This condition is achieved by finding the optimal parameter using CPSO. CPSO reduces the computation time and improves the prediction accuracy. Therefore, the proposed CPSO-LSTM is suitable for temperature forecasting. Although CPSO-LSTM utilizes the sample reduction technique, it requires high processing time for selecting the samples. Some of the environmental factors are temperature, humidity, precipitation, and dew visibility. Weather forecasting is performed by using this parameter and the LSTM network to predict the future values. This approach can predict accurate results only by proper neural network tuning; otherwise, the prediction results may vary.

6 Conclusion

The particle swarm optimization-based long short-term memory network is proposed to forecast the temperature and heat wave by using big data. The proposed CPSO-LSTM outperforms the traditional LSTM algorithm by reducing its computational time. This hyperparameter-based LSTM can predict the temperature accurately by having a 0.250 RMSE value compared with a traditional LSTM value of 0.35. The proposed CPSO-LSTM is advantageous over LSTM. It reduces the computational time for tuning the LSTM hyperparameters (epoch and hidden neuron). The optimal parameter is identified by minimizing the RMSE of LSTM. With these tuned parameters, the CPSO-LSTM can reduce its MAE and RMSE values compared with the LSTM prediction. The results show that the proposed CPSO-LSTM can forecast the temperature accurately. A proper selection of hyperparameters helps to improve the result and reduce computational time. Therefore, the proposed CPSO-LSTM is suitable for forecasting operations on big data. The modifications in LSTM layer operations can improve the prediction accuracy in the future. In the future, we intend to extend this work by analyzing the natural calamity-based methodologies.

Acknowledgement: Two of the authors wish to express gratitude to their organization JSS Academy of Technical Education, Bengaluru-580060, for the encouragement.

Funding Statement: This work was partially funded by the Robotics and the Internet of Things Lab at Prince Sultan University.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. M. Chen and J. R. Wang, "Temperature prediction using fuzzy time series," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 30, no. 2, pp. 263–275, 2000.

[2] I. Maqsood, M. R. Khan and A. Abraham, "An ensemble of neural networks for weather forecasting," *Neural Computing & Applications*, vol. 13, no. 2, pp. 112–122, 2004.

[3] G. Shrivastava, S. Karmakar, M. K. Kowar and P. Guhathakurta, "Application of artificial neural networks in weather forecasting: A comprehensive literature review," *International Journal of Computer Applications*, vol. 51, no. 18, pp. 17–29, 2012.

[4] S. D. Sawaitul, K. P. Wagh and P. N. Chatur, "Classification and prediction of future weather by using backpropagation algorithm-an approach," *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 1, pp. 110–113, 2012.

[5] A. R. Naik and S. K. Pathan, "Weather classification and forecasting using backpropagation feed-forward neural network," *International Journal of Scientific and Research Publications*, vol. 2, no. 12, pp. 1–3, 2012.

[6] I. K. Shereef and S. S. Baboo, "A new weather forecasting technique using a backpropagation neural network with a modified levenberg marquardt algorithm for learning," *International Journal of Computer Science Issues*, vol. 8, no. 6, pp. 153–160, 2011.

[7] G. Papacharalampous, H. Tyralis and D. Koutsoyiannis, "Predictability of monthly temperature and precipitation using automatic time series forecasting methods," *Acta Geophysica*, vol. 66, no. 4, pp. 807–831, 2018.

[8] G. Papacharalampous, H. Tyralis and D. Koutsoyiannis, “Univariate time series forecasting of temperature and precipitation with a focus on machine learning algorithms: A multiple-case study from Greece,” *Water Resources Management*, vol. 32, no. 15, pp. 5207–5239, 2018.

[9] B. M. Yahya and D. Z. Seker, “Designing weather forecasting model using computational intelligence tools,” *Applied Artificial Intelligence*, vol. 33, no. 2, pp. 137–151, 2019.

[10] N. Shivhare, A. K. Rahul, S. B. Dwivedi and P. K. S. Dikshit, “ARIMA-based daily weather forecasting tool: A case study for varanasi,” *MAUSAM*, vol. 70, no. 1, pp. 133–140, 2019.

[11] R. Graf, S. Zhu and B. Sivakumar, “Forecasting river water temperature time series using a wavelet–neural network hybrid modelling approach,” *Journal of Hydrology*, vol. 578, pp. 124115, 2019.

[12] S. Singh, M. Kaushik, A. Gupta and A. K. Malviya, “Weather forecasting using machine learning techniques,” in *Proc. Int. Conf. on Advanced Computing and Software Engineering*, Sultanpur, India, 2019.

[13] M. Bendre and R. Manthalkar, “Time series decomposition and predictive analytics using Map reduce 2019,” *Expert System with Applications*, vol. 116, pp. 108–120, 2019.

[14] H. Lan, C. Zhang, Y. Hong, Y. He and S. Wen, “Day-ahead spatiotemporal solar irradiation forecasting using frequency-based framework,” *Applied Energy*, vol. 247, pp. 389–402, 2019.

[15] L. Ji, Z. Wang, M. Chen, S. Fan, W. Yingchun *et al.,* “How much can AI techniques improve surface air temperature forecast?—A report from AI challenger 2018 Global Weather Forecast contest,” *J. Meteor. Res.*, vol. 33, no. 5, pp. 989–992, 2019.

[16] N. Attoue, I. Shahrour and R. Younes, “Smart building: Use of the artificial neural network approach for indoor temperature forecasting,” *Energies*, vol. 11, no. 2, pp. 395, 2018.

[17] X. Zhang, Q. Zhang, G. Zhang, Z. Nie, Z. Gui *et al.,* “A novel hybrid data-driven model for daily land surface temperature forecasting using long short-term memory neural network based on ensemble empirical mode decomposition,” *International Journal of Environmental Research and Public Health*, vol. 15, no. 5, pp. 1032, 2018.

[18] B. Spencer, O. Alfandi and F. Al-Obeidat, “A refinement of lasso regression applied to temperature forecasting,” *Procedia Computer Science*, vol. 130, pp. 728–735, 2018.

[19] D. N. Fente and D. K. Singh, “Weather forecasting using artificial neural network,” in *Proc. Int. Conf. on Inventive Communication and Computational Technologies*, Coimbatore, India, 2018.

[20] C. Kunjumon, S. S. Nair, P. Suresh and S. L. Preetha, “Survey on weather forecasting using data mining,” in *Proc. Int. Conf. on Emerging Devices and Smart Systems*, Tiruchengode, India, 2018.

[21] M. S. Mahdavinejad, M. Rezvan, M. Barekatain, P. Adibi, P. Barnaghi *et al.,* “Machine learning for internet of things data analysis: A survey,” *Digital Communications and Networks*, vol. 4, no. 3, pp. 161–175, 2018.

[22] B. Wang, J. Lu, Z. Yan, H. Luo, T. Li *et al.,* “Deep uncertainty quantification: A machine learning approach for weather forecasting,” in *Proc. Int. Conf. on Knowledge Discovery & Data Mining*, Anchorage, USA, 2019.

[23] A. Y. Sun and B. R. Scanlon, “How can Big data and machine learning benefit environment and water management: A survey of methods, applications, and future directions,” *Environmental Research Letters*, vol. 14, no. 7, pp. 073001, 2019.

[24] J. B. Jane and E. N. Ganesh, “Big data and internet of things for smart data analytics using machine learning techniques,” in *Proc. Int. Conf. on Computer Networks, Big Data and IoT*, Madurai, India, 2018.

[25] A. Sharma, “Predictive analytics in weather forecasting using machine learning algorithms,” *EAI Endorsed Transactions on Cloud Systems*, vol. 5, no. 14, 2018.

[26] H. K. Tripathy, B. R. Acharya, R. Kumar and J. M. Chatterjee, “Machine learning on big data: A developmental approach on societal applications,” *Big Data Processing Using Spark in Cloud*, pp. 143–165, 2019.

[27] R. Qiu, Y. Wang, B. Rhoads, D. Wang, W. Qiu *et al.,* “River water temperature forecasting using a deep learning method,” *Journal of Hydrology*, vol. 595, pp. 126016, 2021.

[28] S. Lee, Y. S. Lee and Y. Son, “Forecasting daily temperatures with different time interval data using deep neural networks,” *Applied Sciences*, vol. 10, no. 5, pp. 1609, 2020.

[29] A. Ozbek, A. Sekertekin, M. Bilgili and N. Arslan, “Prediction of 10-min, hourly, and daily atmospheric air temperature: Comparison of LSTM, ANFIS-FCM, and ARMA,” *Arabian Journal of Geosciences*, vol. 14, no. 7, pp. 1–16, 2021.

[30] F. Mtibaa, K. K. Nguyen, M. Azam, A. Papachristou, J. S. Venne *et al.,* “LSTM-Based indoor air temperature prediction framework for HVAC systems in smart buildings,” *Neural Computing and Applications*, vol. 32, no. 23, pp. 17569–17585, 2020.

[31] M. C. Sorkun, Ö. D. İncel and C. Paoli “Time series forecasting on multivariate solar radiation data using deep learning (LSTM),” *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 28, no. 1, pp. 211–223, 2020.

[32] P. Maktala and M. Hashemi, “Global land temperature forecasting using long short-term memory network,” in *Proc. Int. Conf. on Information Reuse and Integration for Data Science*, Las Vegas, USA, 2020.

[33] R. Barzegar, M. Taghi Aalami and J. Adamowski, “Short-term water quality variable prediction using a hybrid CNN–LSTM deep learning model,” *Stochastic Environmental Research and Risk Assessment*, vol. 34, pp. 415–433, 2020.

[34] G. Vallathan, A. John, C. Thirumalai, S. Mohan, G. Srivastava *et al.,* “Suspicious activity detection using deep learning in secure assisted living IoT environments,” *The Journal of Supercomputing*, vol. 77, pp. 3242–3260, 2021.

[35] A. Movassagh, A. Alzubi, M. Gheisari, M. Rahimi, S. Mohan *et al.,* “Artificial neural networks training algorithm integrating invasive weed optimization with differential evolutionary model,” *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–9, 2021.

[36] “Average daily temperature archive,” 2021. [Online]. Available: https://academic.udayton.edu/kissock/http/Weather/gsod95-current/INCHENAI.txt (Accessed on 05 August 2021).

[37] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proc. Int. Conf. on Neural Networks*, Perth, Australia, 1995.

[38] A. Sherstinsky, “Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network,” *Physica D: Nonlinear Phenomena*, vol. 404, pp. 132306, 2020.