A Comparative Study of Long Short-Term Memory for Rainfall Prediction in India

Chawngthu Zoremsanga and Jamal Hussain

Abstract Researchers extensively studied the prediction of rainfall due to its significant impact on the environment and the daily lives of individuals. In this study, four LSTM models were applied to predict average monthly rainfall in India, and their performances are compared with a benchmark model found in the literature. Average monthly rainfall data for All-India from 1871 to 2016 was employed for training and testing the four LSTM models. The models are compiled using MSE loss function, and Adam optimization technique was employed as the optimizer. The performance of the four LSTM models was estimated using statistical metrics such as MAE and RMSE. This study found that more numbers of neurons and stacking the LSTM layers can improve the LSTM model performance. The LSTM Model_4 achieved an RMSE of 245.30, whereas the existing benchmark model achieved an RMSE of 251.63.

Keywords Deep learning · LSTM · Rainfall prediction · Stacked LSTM

1 Introduction

Rainfall is the primary source of fresh water, and it significantly impacts every living organism on this earth. It also affects the transportation sectors, agriculture sectors and management of renewable energy. Accurate rainfall prediction is difficult due to the large number of parameters that affect rainfall [1]. Predictions of rainfall methods include physical, statistical and machine learning methods. Numerical weather predictions are used in physical methods, but due to their computational resource and extensive data requirements, they are less feasible for predictions compared to other methods. In the statistical methods, the aim is to reveal the mathematical relationship in the data. Auto regression (AR), moving average (MA), autoregressive integrated moving average (ARIMA) and easonal autoregressive integrated moving average (SARIMA) are commonly used statistical methods. Due to their

C. Zoremsanga (✉) · J. Hussain
Department of Mathematics and Computer Science, Mizoram University, Aizawl, Mizoram, India

S. N. Singh et al. (eds.), *Proceedings of the NIELIT's International Conference on Communication, Electronics and Digital Technology*, Lecture Notes in Networks and Systems 676, https://doi.org/10.1007/978-981-99-1699-3_38

capability for modeling linear and nonlinear data, support vector regression (SVR) and neural networks machine learning methods are widely adopted by researchers [1–3]. Recently, artificial neural networks including recurrent neural network (RNN), deep belief network (DBN), long short-term memory (LSTM), convolutional neural network (CNN) and convolutional long short-term memory (ConvLSTM) gained popularity among researchers. They were applied successfully for time-series prediction, system health management, image recognition, natural language processing and other problems [4–6].

Rainfall prediction has been an active research area for many years. Prediction of rainfall can be categorized based on the methods applied for the prediction and the nature of data, such as meteorological variables, radar imagery or satellite imagery used to predict rainfall [7]. Reference [8] compared RNN and LSTM for the prediction of rainfall in India using a univariate rainfall dataset. They found that LSTM performed better than the RNN model. In this study, four LSTM models are proposed for predicting monthly average rainfall in India, and the results were compared with [8]. All-India monthly average rainfall data for 1871–2016 was used to evaluate the proposed LSTM models.

The paper's organization is given as follows: Survey of literature is given in Sect. 2. The proposed work is discussed in Sect. 3. Section 4 contains the results, and Sect. 5 outlines the conclusion and future directions.

2 Survey of Literature

In this part, we illustrate the prior investigations that have been done on forecasting rainfall in India. Predicting rainfall in India is usually conducted using the weather variables from the meteorological observations such as surface data, upper air observations and data observation from the ocean.

References [9–12] evaluated stacked autoencoder to identify the predictor variable from multiple weather observations for predicting the summer monsoon in India. They implemented regression tree and decision tree coupled with bagging method to predict long-term monsoon in India and showed that their models outperformed the India Meteorological Department (IMD) model.

Machine learning model to predict heavy rainfalls during the monsoon season in Mumbai and Kolkata using surface-level weather parameters, reanalysis data and IMD rainfall data was proposed by [13]. They reduced the features using a Stacked Autoencoder (SAE) and compared the performance of a support vector machine (SVM) with other neural network models.

Reference [14] compared Intensified-LSTM with Holt-Winters, RNN, extreme learning machine (ELM), LSTM and ARIMA models for the prediction of rainfall in India. Meteorological factors such as temperature , humidity, wind speed, evapotranspiration and sunshine data are used for rainfall prediction in the Hyderabad region.

Reference [15] used LSTM deep learning method and sequence-to-sequence (Seq2Seq) method to identify the break and active monsoon period in the central parts of India. The authors' used daily rainfall from June to September (1948–2014) and compared their proposed models with SVM and K-Nearest Neighbor (KNN). They found that LSTM and sequence-to-sequence outperformed SVM and K-Nearest Neighbor methods.

ConvLSTM with Salp-Stochastic gradient descent (S-SGD) algorithm was implemented by [16] to forecast precipitation levels in India using all-India rainfall data and Tamil Nadu state rainfall data during 1901–2015. They compared S-SGD-based ConvLSTM with ConvLSTM, cluster-wise linear regression (CLR) technique, multilayer perception (MLP) classification algorithm and dynamic self-organizing multilayer network inspired by the immune algorithm (DSMIA).

Reference [8] study an LSTM model for predicting monthly rainfall and explored multiple time lags to find the best performing model. They compared the LSTM method with recurrent neural network and applied the models on the Indian homogeneous areas to study the performance of the models. They found that LSTM model achieved better outcome than the RNN model for the prediction of rainfall. They also observed that 12–15 previous time lags produced better results than other time lags.

In this paper, we compared four LSTM models by varying the number of layers and compared the predicted outcome of the models with [8].

3 Proposed Work

3.1 Study Area and Data Source

We collected the all-India monthly rainfall data from the Indian Institute of Meteorology (IITM), Pune [17]. This dataset consists of area-weighted all-India average monthly, seasonal and annual rainfall. The dataset was prepared by [18] from 1871 to 1990 and later updated by [19] for the period of 1991–2016. They used thirty (30) sub-divisional rainfall out of the thirty-six (36) sub-divisions shown in Fig. 1 for preparing this dataset by considering each rain gage station in the sub-divisions.

Figure 2 demonstrates the mean precipitation each month in India from 1871 to 2016 in 10th of mm, and the dataset consists of 1752 months of rainfall data. The statistical description of the dataset for all-India is given in Table 1. The monthly mean rainfall for all-India is 904.94 and a standard deviation of 951.71. The minimum rainfall is 3, whereas the maximum rainfall is found to be 3460. In this study, four LSTM models are trained the using this dataset.

Fig. 1 Meteorological sub-divisions of India. *Source* IITM

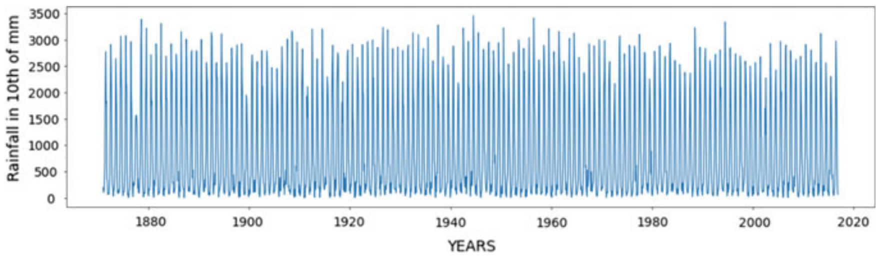

Fig. 2 Average monthly rainfall in India (1871–2016)

3.2 Data Preprocessing

Training and testing data

We separate the rainfall information into two that is for training and for evaluating the performance of the models. For the purpose of this research, 70% of the available information was utilized for the training of the models, and the remaining 30% of the data was employed to evaluate the models' capability in predicting the monthly rainfall.

Table 1 Statistical description of all-India average monthly rainfall data (1871–2016)

Region		All-India
No. of samples		1752
No. of features		1 (rainfall)
Rainfall (in 10th of mm)	Mean	904.94
	Standard deviation	951.71
	Minimum	3
	Maximum	3460
	25th percentile	146.75
	50th percentile	425.0
	75th percentile	1632.5

Normalization

To improve the models' prediction performance, we employed the Min–Max normalization method in the range of 0–1 using the formula given in (1). To prevent information leakage from training data to testing data, the coefficients are estimated using the training dataset only. However, the final predicted data are inverse-transformed to get the prediction to the original scale.

$$d' = d_{\min} + (d_{\max} - d_{\min}) + \left(\frac{x - x_{\min}}{x_{\max} - x_{\min}}\right) \tag{1}$$

The normalized value d' is calculated using (1) given above. In this equation, $d_{\min}$ and $d_{\max}$ are the minimum and maximum values of the data (i.e., 0 and 1, respectively). The symbol x denotes the value to be scaled and, $x_{\min}$ and $x_{\max}$ are the minimum and maximum values of x.

3.3 LSTM

LSTM was introduced to solve the issue of vanishing gradient problem in RNN [20]. LSTM networks have the ability to identify interconnections that span extended time periods within a given dataset, and it has been applied by many researchers in time-series forecasting. This includes a recent study by [21], where they applied LSTM for the prediction of the stock closing price using investors' emotional tendencies. Unlike traditional neural networks that have neurons, LSTM consists of memory blocks. The memory block can be connected in multiple layers, and each block consists of a memory cell and three gates to regulate the cell state.

The basic architecture of LSTM shown in Fig. 3 is made up of three types of gates: forget gate, whose function is to decide which information to retain or reject; the input gate to select which input data is to be used for updating the state of memory

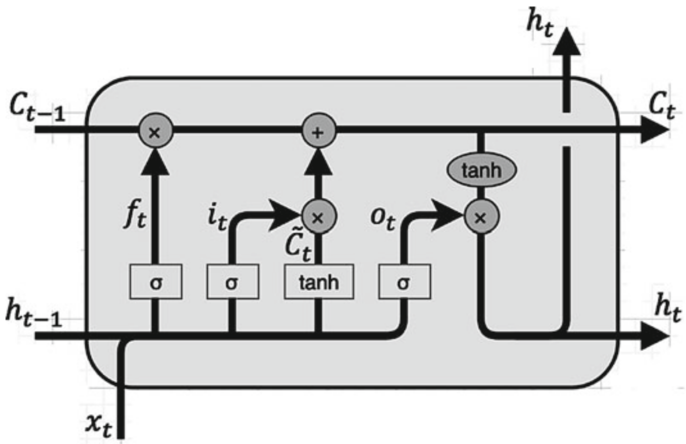

Fig. 3 Basic architecture of LSTM block [23]

cell and the output gate to control the value to output using the value of the input and the memory state [22].

The input sequence received by an LSTM block is passed to the gates, where each gate uses an activation function to decide whether it should activate or not. The gates also have a weight whose value can be learned during the training stage.

The LSTM unit consists of a cell, called the memory unit that has a state c_t at time t. The gates of an LSTM, the input gate i_t, forget gate f_t and the output gate o_t controls the state of the memory unit. A gate also determines the amount of information that passes through, and it is a fully-connected neural network layer that is triggered using a sigmoid function and element-wise multiplication [24]. The following Eqs. (2)–(7) show the flows of information and the working mechanism of the gates:

$$f_t = \sigma\left(W_f.\left[\mathrm{h}_{t-1}x_t\right] + b_f\right) \tag{2}$$

$$i_t = \sigma\left(W_i.\left[\mathrm{h}_{t-1}x_t\right] + b_i\right) \tag{3}$$

$$C'_t = \tanh\left(W_c.\left[h_{t-1}x_t\right] + b_c\right) \tag{4}$$

$$C_t = f_t * C_{t-1} + i_t * C'_t \tag{5}$$

$$o_t = \sigma\left(W_o. = \left[h_{t-1}x_t\right] + b_o\right) \tag{6}$$

$$h_t = o_t * \tanh(C_t) \tag{7}$$

The forget gate f_t, input gate i_t and output gate o_t values are in the range of 0–1, and they are an output of the sigmoid function σ. C_{t-1}, C'_t and C_t represents the old state of the cell, the new candidate value of the cell and the new cell state, respectively. h_t is the output of the LSTM block at time t. W_f, W_i, W_c and W_o are the weight matrices, and b_f, b_i, b_c and b_o are called bias vectors. The * sign indicates element-wise multiplication of two vectors, and an element-wise nonlinear logistic sigmoid and hyperbolic tangent activation functions are written as σ and tanh, respectively.

3.4 Model Development

In this study, four LSTM models were developed for forecasting the average monthly rainfall in India. The compared LSTM models are labeled as Model_1, Model_2, Model_3 and Model_4. The number of past time intervals and the quantity of LSTM cells utilized for the models were identified through experimentation. The number of epochs is tested to the maximum until the data overfits the model and that outputs lowest root mean squared error (RMSE) and mean absolute error (MAE) values.

The LSTM models were constructed using the Adam as the optimization algorithm and mean squared error (MSE) as the loss function. The initial weights are chosen randomly in each execution. Due to the stochastic nature of the algorithm, 10 repetitions were performed on each model, utilizing the same training and evaluation dataset. The average of the ten outputs was calculated to find the final performance of the model.

Model_1 consists of LSTM with one hidden layer with fifty cells and a dense output layer with a single neuron. In Model_2, one hidden dense layer is added to the model, and thus the model consists of one hidden LSTM and dense layer. The hidden LSTM layer has 10 cells, the hidden dense layer consists of 10 neurons, and the output layer is a dense layer with a single neuron. Model_3 comprises two hidden Stacked LSTM layers and a dense output layer. Both the LSTM layers have 10 cells, and the dense output layer has one neuron. In Model_4, there are two LSTM hidden layers with 12 cells each, one hidden dense layer with 12 neurons. The summary of the four LSTM models is given in Table 2.

Table 2 Summary of LSTM models under study

LSTM model	Input timesteps	Model architecture	Epochs
RNN [8]	20	RNN (1)—Dense (1)	500
LSTM [8]	20	LSTM (1)—Dense (1)	500
Model_1	12	LSTM (50)—Dense (1)	400
Model_2	24	LSTM (10)—Dense (10)—Dense (1)	500
Model_3	24	LSTM (10)—LSTM (10)—Dense (1)	500
Model_4	28	LSTM (12)—LSTM (12)—Dense (12)—Dense (1)	500

3.5 Performance Metrics

The performance of the LSTM models was assessed by employing statistical parameters like MAE and RMSE. A lower error score of MAE and RMSE indicates better performance of the model. The final MAE and RMSE values for a single model were calculated using the average of ten MAE and ten RMSE outputs. Equations (8) and (9) show the formula for calculating MAE and RMSE.

$$\mathrm{MAE} = \left(\frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right|\right) \tag{8}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}\right)^2} \tag{9}$$

where y_i is the ith observed rainfall, $\hat{y}_i$ is the ith rainfall predicted by the model and the total number of observed monthly rainfalls is denoted by n.

4 Results and Discussion

Accurate prediction of rainfall is a crucial task due to its influenced in the fields of farming, transportation sectors, disaster management and the daily lives of every living organism. This investigation assessed the effectiveness of four LSTM models Model_1, Model_2, Model 3 and Model_4 for the prediction of average monthly rainfall in India by comparing the impact of increasing LSTM layers. The rainfall data was collected from the Indian Institute of Meteorology (IITM), Pune (IITM Data Archival, 2022). The performance of the studied models- Model_1, Model_2, Model 3 and Model_4 are also compared with the RNN and LSTM model proposed by (Kumar, Singh, Samui, & Jha, 2019). MAE and RMSE are utilized to assess the correctness of the forecasts generated by the models. The number of input-timesteps, no. of epochs and no. of cells for each model are find using trial and error method. The MAE and RMSE are calculated using the default formula provided in the scikit-learn package.

The summary and architectures of the studied LSTM models are shown in Table 2. The comparative performance of the RNN and LSTM model by [8] and the studied models are presented in Table 3. The proposed models' parameters, such as no. of input timesteps and no. of epochs, are found using the trial-and-error method. The RMSE of the proposed LSTM models Model_1, Model_2, Model_3 and Model_4 achieved lower RMSE values compared to the RNN and LSTM models found in the literature [8]. However, the higher values in MAE could be due to the difference in the formula used in this study. The plot of the MSE loss for the LSTM models during the training and validation is shown in Fig. 4.

Table 3 Performance analysis of proposed LSTM models for the prediction of rainfall

LSTM model	Input timesteps	No. of epochs	Loss function	Optimizer	MAE	RMSE
RNN [8]	20	500	MSE	SGD	279.6	261.7
LSTM [8]	20	500	MSE	SGD	169.35	251.63
LSTM Model_1	12	400	MSE	Adam	174.35	250.47
LSTM Model_2	24	500	MSE	Adam	170.60	246.36
LSTM Model_3	24	500	MSE	Adam	169.89	246.48
LSTM Model_4	28	500	MSE	Adam	168.64	245.30

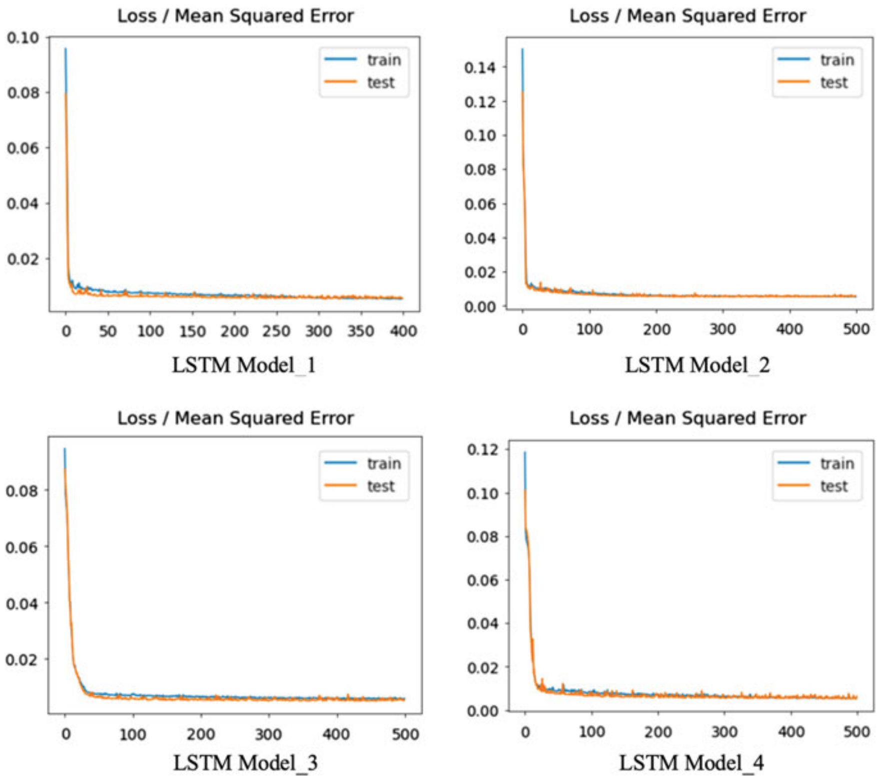

Fig. 4 Performance (MSE) Plot of LSTM models for training and validation

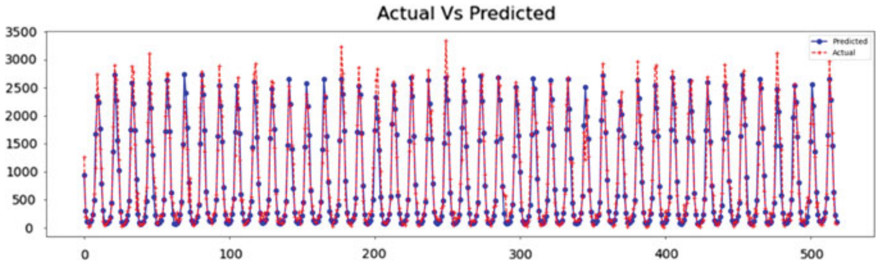

Fig. 5 Prediction result of Model_1

Among the proposed LSTM models, Model_4 has achieved the lowest RMSE value, 245.30 compared to Model_1: 250.47, Model_2: 246.36 and Model_3: 246.48. Analysis of the performance metrics of each model also shows that including more previous timesteps also leads to better model performance. The plot of the prediction results for Model_1, Model_2, Model_3 and Model_4 is shown in Figs. 5, 6, 7 and 8. From this figure, it can also be observed that Model_4 generalized the testing data more accurately compared to other models.

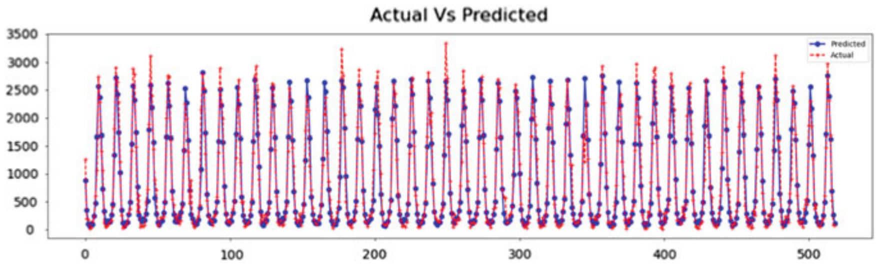

Fig. 6 Prediction result of Model_2

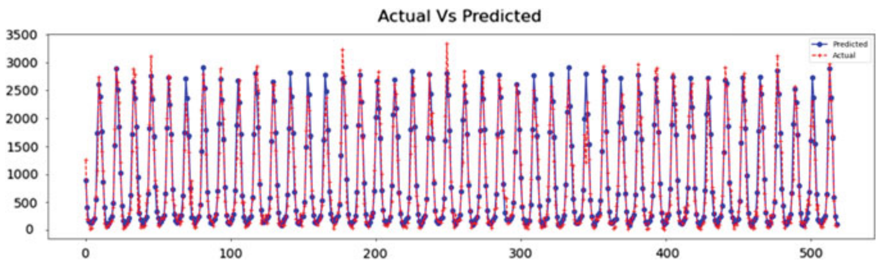

Fig. 7 Prediction result of Model_3

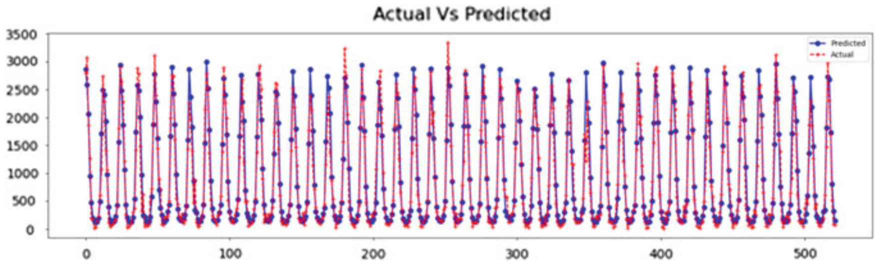

Fig. 8 Prediction result of Model_4

5 Conclusions

This study applied four LSTM models to predict monthly average rainfall in India during 1871–2016. The achievement of the four LSTM models was estimated using statistical metrics such as MAE and RMSE. LSTM Model_4 has achieved the lowest RMSE value of 245.30 compared to Model_1: 250.47, Model_2: 246.36 and Model_3: 246.48. This shows that stacked LSTM can predict rainfall series, and increasing the timesteps also improved the model's performance. The LSTM models in this study are also compared with the RNN and LSTM models of [8], and they all achieved lower RMSE. This shows that using more numbers of neurons and stacking the LSTM layers can improve the model performance. Future work will focus on finding the prediction performance of Bi-LSTM and different optimizers for further improvement of the prediction performance.

References

1. Dash Y, Mishra SK, Panigrahi BK (2018) Rainfall prediction for the Kerala state of India using artificial intelligence approaches. Comput Electr Eng 70:66–73. https://doi.org/10.1016/j.compeleceng.2018.06.004
2. Manzato A (2007) Sounding-derived indices for neural network based short-term thunderstorm and rainfall forecasts. Atmos Res 83:349–365. https://doi.org/10.1016/j.atmosres.2005.10.021
3. French MN, Krajewski WF, Cuykendall RR (1992) Rainfall forecasting in space and time using a neural network. J Hydrol (Amst) 137:1–31. https://doi.org/10.1016/0022-1694(92)90046-X
4. Shen C (2018) A transdisciplinary review of deep learning research and its relevance for water resources scientists. Water Resour Res 54:8558–8593. https://doi.org/10.1029/2018WR022643
5. Khan S, Yairi T (2018) A review on the application of deep learning in system health management. Mech Syst Signal Process 107:241–265. https://doi.org/10.1016/j.ymssp.2017.11.024
6. Nash W, Drummond T, Birbilis N (2018) A review of deep learning in the study of materials degradation. Npj Mater Degrad 2. https://doi.org/10.1038/s41529-018-0058-x
7. Hussain J, Zoremsanga C (2021) A survey of rainfall prediction using deep learning. In: 3rd international conference on electrical, control and instrumentation engineering (ICECIE). IEEE, pp 1–10
8. Kumar D, Singh A, Samui P, Jha RK (2019) Forecasting monthly precipitation using sequential modelling. Hydrol Sci J 64:690–700. https://doi.org/10.1080/02626667.2019.1595624

9. Saha M, Mitra P, Nanjundiah RS (2016) Predictor discovery for early-late Indian Summer Monsoon using stacked autoencoder. Proc Comput Sci 565–576
10. Saha M, Mitra P, Nanjundiah RS (2016) Autoencoder-based identification of predictors of Indian monsoon. Meteorol Atmos Phys 128:613–628. https://doi.org/10.1007/s00703-016-0431-7
11. Saha M, Mitra P, Nanjundiah RS (2017) Deep learning for predicting the monsoon over the homogeneous regions of India. J Earth Syst Sci 126. https://doi.org/10.1007/s12040-017-0838-7
12. Saha M, Santara A, Mitra P, Chakraborty A, Nanjundiah RS (2021) Prediction of the Indian summer monsoon using a stacked autoencoder and ensemble regression model. Int J Forecast 37:58–71. https://doi.org/10.1016/j.ijforecast.2020.03.001
13. Gope S, Sarkar S, Mitra P, Ghosh S (2016) Early prediction of extreme rainfall events: a deep learning approach. In: Perner P (ed) Advances in data mining. Applications and theoretical aspects. ICDM. Springer International Publishing, Cham, pp 154–167
14. Poornima S, Pushpalatha M (2019) Prediction of rainfall using intensified LSTM based recurrent neural network with weighted linear units. Atmosphere (Basel) 10. https://doi.org/10.3390/atmos10110668
15. Viswanath S, Saha M, Mitra P, Nanjundiah RS (2019) Deep learning based LSTM and SeqToSeq models to detect monsoon spells of India. In: Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics). Springer Verlag, pp 204–218
16. Manoj O, Ananth JP (2020) MapReduce and optimized deep network for rainfall prediction in agriculture. Comput J 63:900–912. https://doi.org/10.1093/comjnl/bxz164
17. Data Archival2. https://tropmet.res.in/static_pages.php?page_id=53
18. Parthasarathy B, Kothawale DR (1995) Monthly and seasonal rainfall series for all-India, homogeneous regions and meteorological subdivisions: 1871–1994
19. Kothawale DR, Rajeevan M (2017) Monthly, seasonal and annual rainfall time series for all-India, homogeneous regions and meteorological subdivisions: 1871–2016
20. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9:1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
21. Jin Z, Yang Y, Liu Y (2020) Stock closing price prediction based on sentiment analysis and LSTM. Neural Comput Appl 32:9713–9729. https://doi.org/10.1007/s00521-019-04504-2
22. Abdel-Nasser M, Mahmoud K (2019) Accurate photovoltaic power forecasting models using deep LSTM-RNN. Neural Comput Appl 31:2727–2740. https://doi.org/10.1007/s00521-017-3225-z
23. Understanding LSTM Networks—Colah's blog. http://colah.github.io/posts/2015-08-Understanding-LSTMs/
24. Silva AQB (2019) Predicting cervical cancer with metaheuristic optimizers for training LSTM. In: Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics). Springer Verlag, pp 642–655