Research Article

Mustain Billah*, Md. Nasim Adnan, Mostafijur Rahman Akhond, Romana Rahman Ema, Md. Alam Hossain, and Syed Md. Galib

Rainfall prediction system for Bangladesh using long short-term memory

https://doi.org/10.1515/comp-2022-0254
received January 19, 2022; accepted August 5, 2022

Abstract: Rainfall prediction is a challenging task and has extreme significance in weather forecasting. Accurate rainfall prediction can play a great role in agricultural, aviation, natural phenomenon, flood, construction, transport, etc. Weather or climate is assumed to be one of the most complex systems. Again, chaos, also called as "butterfly effect," limits our ability to make weather predictable. So, it is not easy to predict rainfall by conventional machine learning approaches. However, several kinds of research have been proposed to predict rainfall by using different computational methods. To accomplish chaotic rainfall prediction system for Bangladesh, in this study, historical data set-driven long short term memory (LSTM) networks method has been used, which overcomes the complexities and chaos-related problems faced by other approaches. The proposed method has three principal phases: (i) The most useful 10 features are chosen from 20 data attributes. (ii) After that, a two-layer LSTM model is designed. (iii) Both conventional machine learning approaches and recent works are compared with the LSTM model. This approach has gained 97.14% accuracy in predicting rainfall (in millimeters), which outperforms the state-of-the-art solutions. Also, this work is a pioneer work to the rainfall prediction system for Bangladesh.

Keywords: LSTM, deep learning, PCA, machine learning, rainfall prediction

1 Introduction

Bangladesh is primarily an agricultural nation, and the sector's contribution to rapid economic growth is crucial. In order to guarantee long-term food security for humans, it is crucial to create a lucrative, sustainable, and environmentally friendly agricultural system [1]. As a result, accurate rainfall forecasting is an essential tool for the economy's improved development. Rainfall predictions are used in weather forecasting, date fixing of crop plantation, flood prediction, and many other sectors [2]. So undoubtedly, rainfall prediction holds a huge significance and importance.

In the past, monthly rainfall predictions were done using linear regression models [3–5]. Later application of random forest regression models made predictions more accurate [6,7]. Different popular approaches including Naive Bayes (NB) [8], decision tree [9], random forest and some other classification techniques [10–12] have also been used for the same purpose. Other attempts and research works have utilized various ways ranging from pure probabilistic mathematical approach to advanced techniques of machine learning. However, using raw probabilistic formulas cannot yield high enough accuracy due to the complex nature of the datasets [13]. Therefore, dealing with optimized and standardized data along with implementing advanced neural networks will obviously yield better accuracy in predictions.

Predicting rainfall is a time-series problem which can be solved with great efficiency using long short-term memory (LSTM) networks [14]. LSTM is a special branch of recurrent neural networks (RNNs). Previous predictions in this field have not been involved heavily with LSTM network [15]. As LSTM network implementation is relatively advanced technique, it is an optimal choice for quality prediction. Machine learning predictions require huge amount of data for high-end predictions, while it must also be ensured that a significant number of features are involved in the prediction algorithm. If the size of the dataset is not enough, then the quality of prediction does not reach maximum threshold [16].

* **Corresponding author: Mustain Billah,** Department of Computer Science and Engineering, Jashore University of Science and Technology, Jashore, Bangladesh, e-mail: mu.billah@just.edu.bd
Md. Nasim Adnan, Mostafijur Rahman Akhond, Romana Rahman Ema, Md. Alam Hossain, Syed Md. Galib: Department of Computer Science and Engineering, Jashore University of Science and Technology, Jashore, Bangladesh

One of the major challenges in processing real-life data is its high number of dimensions. In order to cut down dimensions of data and thereby to avoid overfitting [17], we have used principal component analysis (PCA), in which unnecessary dimensions (or features) are trimmed off and projected onto other vital dimensions. Thus, we have used optimized data as parameters of prediction in our LSTM network [18]. After feature selection we have applied cross validation technique. In the process of training and testing, the optimized data are then feed forwarded through the input layers to fully connected layers of deep LSTM–LSTM network. Finally, after applying the sigmoid function, we derived the predicted quantity of rainfall followed by measuring the accuracy of the test dataset predictions.

The main contributions of this work are as follows:

- To the best of our knowledge, this is one of the pioneer works on Bangladeshi rainfall prediction system.
- LSTM–LSTM network was built for fast and highly accurate classification.
- The result has been compared with different machine learning approaches.
- Instead of taking all the features, the most useful ten features had been chosen for enhancing execution.

The next part of this article is organized as follows. In Section 3, the proposed method and structure is described. Section 3 has three subsections where data set and feature reduction are discussed. Results are analyzed in Section 4. This article is concluded in Section 5.

2 Literature review

Various data mining techniques have been used for rainfall prediction. Numerous research papers are found in the literature.

In ref. [19], support vector machine (SVM), artificial neural networks (ANN), and Adaptive Neuro Fuzzy Inference System (ANFIS) have been used. ANFIS avoided noise in data by using varying lags of input and SVM dealt with out-of-pattern instances in the dataset. In another research [20], the team in Malaysia worked with random forest, SVM, NB, ANN, and decision tree. They obtained data from various weather stations of Malaysia (Selangor). The noise and missing values in that dataset were fixed using ANFIS and other pre-processing techniques. Random forest algorithm was used to utilize small training dataset against large amount of test data. The study of Chatterjee et al. [21] is based on a survey which examined the Neural Network architectures used for rainfall prediction over the last 25 years. They implemented SVM, multilayer perceptron (MLP), BPN, RBFN, and SOM for forecasting. In the Lahore rainfall dataset [22], a combination of SVM, NB, k-Nearest Neighbor (kNN), decision tree (J48) and MLP has been used to calculate $F1$-scores. F-1 values represent correlation and balance between precision and recall. Higher F-1 values indicate that precision and recall are less complementary. Research team in ref. [23] implemented Back Propagation, Radial Basis Function, and ANN and obtained quality prediction on monthly rainfall. The dataset was from Coonoor region in Nilgiris district (Tamil Nadu). Mean square error was taken as the performance measure. They obtained smaller mean square error using Radial Basis Function Neural Network. In ref. [24], usage of linear regression method was notable. This article also involved a new feature named “rainfall pace” and correlated it with various seasonal crops such as rabi, Kharif, zaid and then predicted future rainfall. The dataset was sorted and grouped according to crop name, season, and year.

In ref. [25], a combination of classical ANN and Genetic Algorithm has produced a hybrid system, which is intelligent enough to make plausible predictions. The purpose of using the Genetic Algorithm was to better organize the input layer, the connection and communication among input nodes, the output layers, and to train the network more efficiently. In ref. [26], using only ANNs, 1-month and 2-month forecasting models have been built that predicted rainfall. Multiple stations in North India have contributions to the dataset and the data records were on past 141 years. Along with both Feed Forward and Back Propagation Neural Network, Levenberg–Marquardt training function has been used. Performance evaluation has been done by mean square error, regression analysis, and magnitude of relative error. Surprisingly, the results proved that the forecasting model for 1 month can predict more accurately the rainfall than that of 2 month.

In ref. [27], authors worked with a time series dataset containing data on Monsoon Rainfall of Summer in India (based on monthly and seasonal time periods of 1871 till 2014). Due to the unpredictable and dynamic nature of monsoon rainfall, entropy theory, fuzzy set, and ANN were used. The fuzzy set theory handled uncertainties and made transition from one inference to another smooth using degrees. In the modified entropy computational concept, the input was treated as a degree of membership into the entropy function (also called Fuzzy Information-Gain [FIG]). The ANN then defuzzified every fuzzified rule. Finally, the output of FIG of every fuzzy-set was fed

Table 1: A comparative analysis of works found in the literature deploying rainfall or weather forecasting

Paper	Dataset	Materials	Merits	Demerits
[20]	Used nearly about 4 years data from Meteorological Department and Drainage and Irrigation Department, Malaysia	Utilized 5 classifiers and claimed random forest to obtain better result	Comparative analysis of different classifiers	Less Precision and Recall value, small training dataset, small feature set
[24]	Dataset consists of past year's rainfall of India	Linear regression	Help farmers to make a correct decision to harvest a particular crop	Based on only 1 feature; no experimental result found;
[26]	Monthly time series rainfall data of North India, collected by Indian Meteorological Department, Pune	ANN	Long time series data set	Very small feature set; for only 1-month and 2-month ahead prediction;
[27]	Indian summer monsoon rainfall time series data	Fuzzy entropy-neuro-based expert system	Time series data are analyzed with statistical parameters	Less accuracy; covers very smaller region
[22]	Conventional weather stations (CWS) in eight Brazilian states	ANN	One of the pioneer work for Brazilian rainfall forecast	Less accuracy; not stable prediction for whole year;
[21]	Dataset obtained from Dumdum weather station	Hybrid neural network	Feature selection; hybrid neural network	Less accuracy; only for a smaller region;

into the ANN. Evaluation of performance was done through root mean square error (RMSE), standard deviations (SDs), correlation coefficient (CC), and performance parameter (PP). This integrated combination worked better than many other models.

Table 1 demonstrates a comparative analysis showing the merits and demerits as long as methods, datasets, and materials that the literature have deployed for rainfall or weather forecasting.

From Table 1, it is seen that most of the works are for different specific regions having some major drawbacks such as very small data set, small feature set, and less accuracy. However, in this work, two-layer LSTM method has been developed [28–30] for predicting rainfall in Bangladesh. It solves the backflow problem [31,32] found in other works. A larger dataset containing 20 features (most prominent 10 features) is used. Proposed work can predict rainfall in advance for any season and for any region of Bangladesh.

3 Proposed structure

In this work (Algorithm 1), we have made rainfall prediction by the LSTM model. Rainfall data has been collected and preprocessed. Also, we have normalized the dataset for making it perfect for applying in LSTM. After normalizing the dataset, a feature selection technique, PCA is applied. Initially, the dataset contained totally 20 features. As all the features may not be suitable for every purposes, feature selection method was applied on the dataset. However, ten most effective features are selected. LSTM is used as a deeper learning-based network. For training the LSTM model, 10-fold cross validation method was applied for splitting the training and testing dataset properly. Two levels of LSTM were applied before the fully connected layer. As output, the model predicts amount of rainfall in millimeters. Flow diagram of the proposed work is as depicted in Figure 1.

Algorithm 1: Rainfall prediction Pseudo code.

```
Result: Rainfall in Millimetre
Rainfall Dataset;
Load attributes of the dataset;
[rank;weigths] = PCA(features.target); //Apply PCA effective features;
for $i \leftarrow 1$ to 10 do
  | $features^{new}$(:,i) = features(:,rank(i)); // Select top 10 features
```

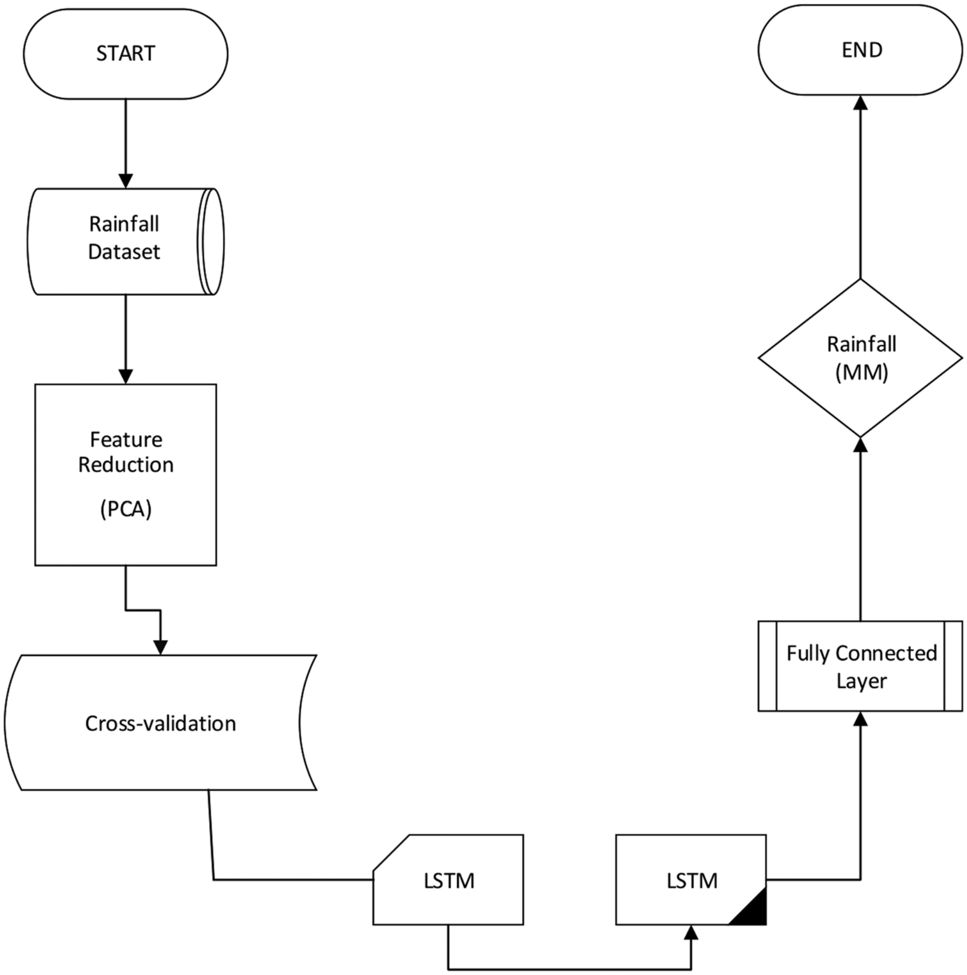

Figure 1: Flow diagram of the proposed system.

```
end
Cross Validation;
$acc^{mean}$ = 0; // Determine the obtain value;
for $j \leftarrow 1$ to 30 do
    cross-validate the data using two LSTM layers .;
    acc=0;
    for $i \leftarrow 1$ to $L$ do
        if $prediction(i) = response(i)$then
            acc = acc + 1;
        end
    end
    $Accuracy^{mean} = Accuracy^{mean} + \frac{Accuracy}{L}$
end
```

3.1 Dataset collection

The rainfall data set [33] was used in this research. The data have been accumulated from the Bangladesh Meteorological Department (BMD), which is a government-owned department. However, the data are not organized. The organized data can be found on kaggle [33]. This dataset comes with 2,366 rows and 20 features (Table 2). In this table, each of the features with weight value and ranking are listed. Daily information of rainfall data (in mm) ranging from 2012 to 2018 are recorded here. This dataset has 20 features (year, month, day, temperature [temp] high, temp average [avg], temp low, DP high, DP avg, DP low, humidity high, humidity avg, humidity low, SLP high, SLP avg, SLP low, visibility high, visibility avg, visibility low,

Table 2: Rainfall dataset

ID	Features	Weight	Weight rank
1	Humidity high	0.020688362	10
2	Humidity avg	0.0192508	11
3	Humidity low	0.009211251	12
4	Visibility high	0.00273289	16
5	Visibility avg	0.002168759	17
6	Visibility low	0.000660308	18
7	Year	0.460823242	1
8	Month	0.123572965	2
9	Day	0.080288179	3
10	Wind high	0.00055288	19
11	Wind avg	0.000372772	20
12	SLP low	0.004164449	15
13	SLP high	0.005305575	13
14	SLP avg	0.004197224	14
15	DP avg	0.034320957	8
16	DP high	0.037307926	7
17	DP low	0.030278182	9
18	Temp high	0.062790121	4
19	Temp low	0.050063972	6
20	Temp avg	0.051249269	5

wind high, wind avg). All data in the dataset are numerical value, so we do not need any embeddings. But, we needed to clean up and standardize the data. All of the information in the dataset are not needed for this study. This is why, the information on whether a particular day was foggy or stormy was omitted. Only the amount of rainfall was considered. The redundant features caused overfitting in the proposed model. For better performance, in the feature reduction [34] step, the number of features was reduced by using PCA. After applying PCA, ten most weighted features were selected (Table 3).

3.2 Feature reduction

Often in deep learning classification problems, ultimate classification is performed based on various factors. These factors are radically variables which are named as features. If the number of features is higher, it is hard to work with these features. The features sometimes are correlated, and sometimes unnecessary. Therefore dimensionality reduction algorithms are needed. In this case, we have used PCA [35–37] to the dataset to get the weight of all features. This weight represents a feature with a higher correlation with class. From the 20 features, we kept ten and removed others (Table 3). The responsibility of the PCA algorithm is to summarize 20 dimensions onto 10 principal dimensions, which are fed into the proposed model for prediction.

Table 3: Selected features from the rainfall dataset

ID	Features	Weight	Weight Rank
1	Year	0.460823242	1
2	Month	0.123572965	2
3	Day	0.080288179	3
4	Temp high	0.062790121	4
5	Temp avg	0.051249269	5
6	Temp low	0.050063972	6
7	Dp high	0.037307926	7
8	Dp avg	0.034320957	8
9	Dp low	0.030278182	9
10	Humidity high	0.020688362	10

3.3 Working process of LSTM

Instead of randomly splitting the dataset into training and testing, 10-fold cross-validation technique has been used. Dividing the whole dataset into 10-folds, the model is fitted using the first fold for testing and others for training. After that, second fold is used for testing and remaining for training the model. This process will repeat until 10th iteration. In every iteration, it gives an error estimation. Then mean is applied to all error estimation.

LSTM has been used for a deeper learning-based classifier (Figure 2). Two LSTM layers are used for enhancing efficiency. Both LSTM layers have 50 hidden neurons units. In the first LSTM, dropout layer is applied. Dropout technique was used to prevent a model from overfitting. After that, this method approximates training a huge sum of networks with different architectures in parallel. In the training phase, some outputs are randomly "dropped out." This has the impact of creating the layer look-like and be treated-like a layer with a special range of nodes and property to the previous layer. In this study, the dropout technique has been used for both LSTM layers. In the first LSTM, 50% dropout was used, while for the second LSTM layer it was 20%.

In every LSTM layer, batch normalization has been used (Figure 3). Batch normalization is a technique for improving the speed, performance, and stability of deep learning-based architecture. Otherwise, there is probability of some numerical data point in the dataset such that one is very high while others might be very low. The data need to be on the same scale for reasonable comparison and analysis. The larger data point in this non-normalized dataset can cause instability in neural networks, which may cause an imbalanced gradient problem. To avoid this, batch normalization technique was used. Subsequently, a two-layer LSTM model had been developed. Hidden neuron units are set to be 50 for both LSTM layers

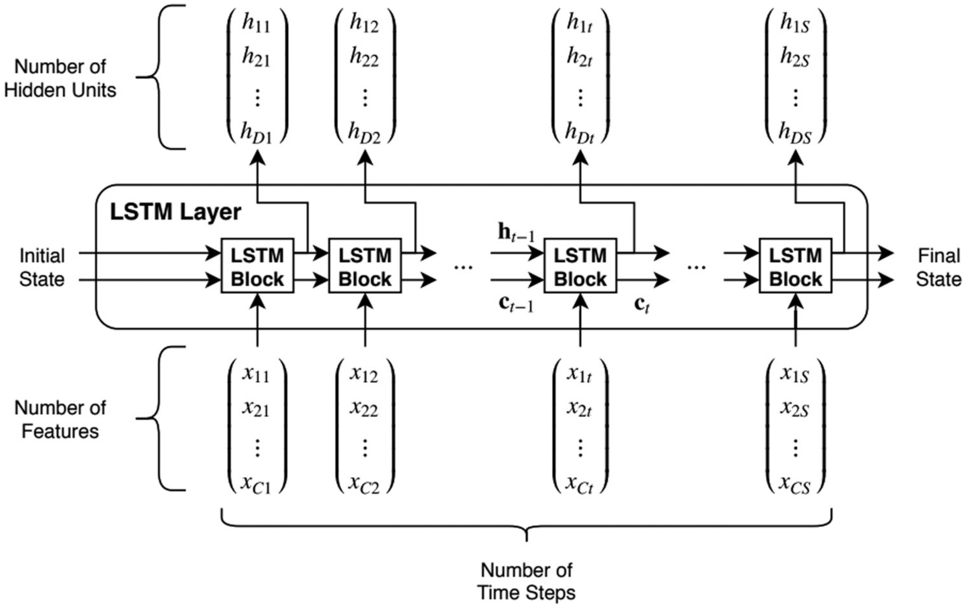
Number of
Hidden Units
h_{11} h_{21} h_{D1}
h_{12} h_{22} h_{D2}
h_{1t} h_{2t} h_{Dt}
h_{1S} h_{2S} h_{DS}
LSTM Layer
Initial
State
LSTM Block
$\mathbf{h}_{t-1}$
$\mathbf{c}_{t-1}$
$\mathbf{c}_t$
Final
State
Number of
Features
x_{11} x_{21} x_{C1}
x_{12} x_{22} x_{C2}
x_{1t} x_{2t} x_{Ct}
x_{1S} x_{2S} x_{CS}
Number of
Time Steps

Figure 2: LSTM network.

and the “epoch” is set to be 30. Table 4 contains all the mathematical formula and measurements deployed in this work.

4 Results and discussion

4.1 Experimental setup

A server computer RAM 16GB, 2-cor Intel(R) Xeon(R) CPU@ 2.20GHz processor and GPU: 1xTesla K80 12GB GDDR5 VRAM was used.

4.2 Comparison with different classifiers

Various machine learning approaches have been deployed to compare with the proposed LSTM approach such as KNN, logistic regression, SVM, random forest, NB, neural network, and LSTM. KNN algorithm is a simple classification algorithm, which is the most used learning algorithm in machine learning. From dataset, kNN uses several classes to predict the result of the output [38]. Logistic regression is effective for the condition that gives the result in the form of binary regression [10]. SVM is a supervised machine learning algorithm with associated learning algorithms that analyzed data used for regression and classification problems. It transforms the data and these

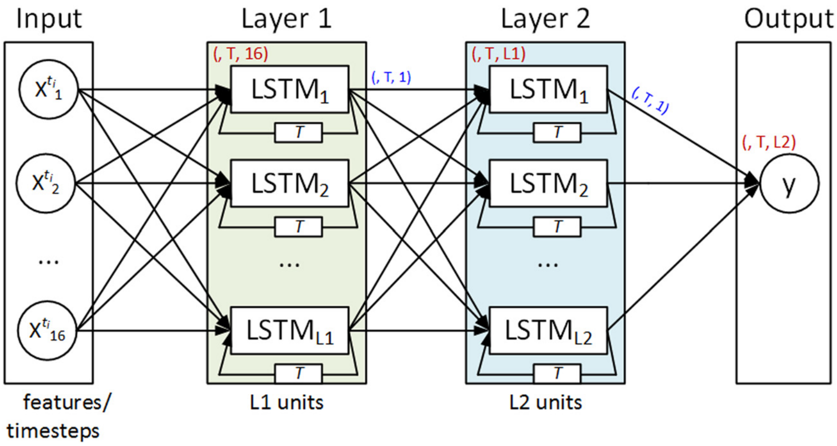

Figure 3: Structure of double LSTM layers.

Table 4: Deployed mathematical formula and measurements

Formula used in PCA	If we use PCA for dimensionality reduction, we construct a $d * k$-dimensional transformation matrix W that allows us to map a sample vector x onto a new k-dimensional feature subspace that has fewer dimensions than the original d-dimensional feature space: $x = [x_1, x_2, \ldots, x_d], x \epsilon R^d, W \epsilon R^{d*k}$ (1) $z = [z_1, z_2, \ldots, z_k], z \epsilon R^k$ (2)
Formula used in LSTM	LSTM network maps an input sequence $x = (x_1, x_2, \ldots, x_T)$ to and output sequence $y = (y_1, y_2, y_3, \ldots, x_T)$ calculating network unit activations using the following equations iteratively from $t = 1$ to T as follows: $i_t = \sigma(W_i + U_i h_{t-1} + b_i)$ (3) $z_t = \tanh(W_z x_t + U_z h_{t-1} + b_z)$ (4) $f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$ (5) $C_t = i_t * z_t + f_t * C_{t-1}$ (6) $o_t = \sigma(W_o x_t + U_o h_{t-1} + V_o C_t + b_o)$ (7) $h_t = 0_t * \tanh(C_t)$ (8) Here, W_t, W_z, W_f, W_o, U_i, U_z, U_f, U_o are model parameters which are estimated during model training; σ and tanh are activation functions; b is bias
Performance measurements	To assess the performance of proposed methods against other classifiers the following measures have been adopted: $\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{FP} + \text{TN}},$ (9) where TP is the True Positive rate, TN is the True Negative rate, FP is the False Positive rate, FN is the False Negative rate

transformed data are used to find the feasible outputs [11]. Random forest model is also known as the random forest tree, which is a popular machine learning classifier. It uses a decision tree for training and outputs the class. It gives the result of the individual trees followed by providing all tress a mean result for prediction [9]. NB algorithm is called "naive," because it makes an opinion that all properties are free from each other. Notwithstanding NB

Table 5: Result of different classifier on same dataset

Classifier	Accuracy (%)
KNN	76.8
Logistic regression	76.6
SVM	76.7
Random forest	71.6
NB	63.3
Neural network	76.9
Proposed method (LSTM with PCA)	97.14

Table 6: Comparison with other works

Paper	Country	Dataset	No. of features	Tool	Evaluation metrics
[20]	Malaysia	Malaysia Meteorological Department and Malaysia Drainage and Irrigation Department spanning from Jan 2010 until April 2014	5 features	Random forest	Precision: 0.715, Recall: 0.738, F-measure: 0.723
[24]	India	Dataset consists of past year's rainfall of India	1 feature	Linear regression	Not specified
[26]	India	monthly time series rainfall data of North India for the period 1871 to 2012 (141 years) collected by Indian Meteorological Department, Pune	3 features	ANN	Accuracy: 93.9%
[27]	India	Indian summer monsoon rainfall time series data	Time series data	Fuzzy entropy-neuro-based expert system	RMSE 6.24%
[22]	Brazil	CWS in eight Brazilian states	7 features	ANN	Accuracy: 78% on summer, 71% on winter 62% on spring and 56% on autumn
[21]	West Bengal	Dataset obtained from Dumdum weather station	8 features	Hybrid neural network	Accuracy: 89.54%
Proposed method	Bangladesh	BMD	10 features	LSTM with PCA	Accuracy = 97.14%

classifier works remarkably well in the machine learning approach [8]. The neural network is an algorithm inspired by the structure of the human brain. Every node is fully connected. A neural network learns to accomplish tasks by analyzing cases. The neural network takes data and trains themselves to identify the pattern in these data after that it predicts the output for a new set of comparable data [12].

Results of different classifier on the same dataset are shown in Table 5. The deep learning LSTM gives better results than conventional machine learning approaches. The two-layer LSTM model obtains better outcomes. All other machine learning approaches have obtained accuracy of 76% with feature selection with the best weights by the PCA method, where the double LSTM layer gives the best result of 77.14%. In other classifiers, the Rainfall dataset may face some overfitting problems. But in both LSTM layers, we used the Dropout technique to avoid overfitting. So LSTM gives more accurate results. Moreover, Rainfall data set has 20 attributes. The PCA method is applied for feature selection. By applying PCA ten best features are selected with high weights. Then two LSTM layers are designed with the neuron of 50 units. This designed architecture achieves the best results compared to any other classifier. It has obtained an accuracy value of 97.14%.

4.3 Comparison with other works

In deep learning, LSTM has been used as a special RNN. LSTM can classify, process, and make predictions based on time series data. Prediction methods typically face some problems such as insufficient capacity of memory, occurrence of vanishing gradient in the network, increased prediction error etc. In this study, two LSTM layers are used for the deeper network as the more hidden layer in the model, the more accurate prediction it gives. By employing the LSTM network, it overcomes capacity insufficiency. Again, by multiplying input sequence to the layers of LSTM it mitigates the vanishing gradient, while it reduces the prediction error by using optimizers. Performance of the proposed approach has been compared with other works found in the literature in terms of dataset, country, number of features, evaluation metrics, and tools or platforms (Table 6).

5 Conclusion and future work

In this work, a two-layer LSTM model has been applied for the Bangladeshi rainfall prediction system. Weights of features are calculated with the PCA method. Furthermore, the most weighted ten features are used. The proposed method has been compared with the performance of the traditional machine learning classifiers, such as KNN, Logistic Regression, SVM, random forest, NB and neural network. The accuracy of the Rainfall dataset had been compared with different classifiers. The best results were achieved by neural network 76.9% and KNN 76.8%. Proposed method obtained 97.14% accuracy. The result obtained by the proposed method is higher than every classifier applied for this dataset. In future works, the proposed system will be examined by larger data sets. Meanwhile, higher result can be obtained by using the Bidirectional LSTM layer. Additionally, a web application will also be formulated.

Conflict of interest: The authors declare that they have no conflict of interest.

Data availability statement: The datasets generated during and/or analysed during the current study are available in the “https://www.kaggle.com/redikod/historical-rainfall-data-in-bangladesh”.

References

[1] E. C. Stephens, A. D. Jones, and D. Parsons, “Agricultural systems research and global food security in the 21st century: An overview and roadmap for future opportunities,” *Agricult. Sys.*, vol. 163, pp. 1–6, 2018.

[2] D. Bhandari and A. Dixit, “Missed Opportunities in Utilization of Weather Forecasts: An Analysis of October 2021 Disaster in Nepal,” ISET Nepal Publication, Nepal, 2022.

[3] F. Mekanik, M. A. Imteaz, S. Gato-Trinidad, and A. Elmahdi, “Multiple regression and artificial neural network for long-term rainfall forecasting using large scale climate modes,” *J. Hydrol.*, vol. 503, pp. 11–21, 2013.

[4] S. Prabakaran, P. Naveen Kumar, and P. Sai Mani Tarun, “Rainfall prediction using modified linear regression,” *ARPN J. Eng. Appl. Sci.*, vol. 12, no. 12, pp. 3715–3718, 2017.

[5] T. DelSole and J. Shukla, “Linear prediction of Indian monsoon rainfall,” *J. Climate*, vol. 15, no. 24, pp. 3645–3658, 2002.

[6] P.-S. Yu, T. C. Yang, S. Y. Chen, C. M. Kuo, and H. W. Tseng, “Comparison of random forests and support vector machine for real-time radar-derived rainfall forecasting,” *J. Hydrol.*, vol. 552, pp. 92–104, 2017.

[7] J. Diez-Sierra and M. delJesus, “Subdaily rainfall estimation through daily rainfall downscaling using random forests in Spain,” *Water*, vol. 11, no. 1, p. 125, 2019.

[8] H. Zhang, J.-X. Ma, C.-T. Liu, J.-X. Ren, and L. Ding, “Development and evaluation of in silico prediction model for drug-induced respiratory toxicity by using Naïve Bayes classifier method,” *Food Chem. Toxicol.*, vol. 121, pp. 593–603, 2018.

[9] X. Zhu, X. Du, M. Kerich, F. W. Lohoff, and R. Momenan, “Random forest based classification of alcohol dependence patients and healthy controls using resting state MRI,” *Neurosci. Lett.* vol. 676, pp. 27–33, 2018.

[10] G. Manogaran and D. Lopez, “Health data analytics using scalable logistic regression with stochastic gradient descent,” *Int. J. Adv. Intell. Paradigms*, vol. 10, no. 1–2, pp. 118–132, 2018.

[11] G. M. Foody and A. Mathur, “Toward intelligent training of supervised image classifications: directing training data acquisition for SVM classification,” *Remote Sensing of Environment*, vol. 93, no. 1–2, pp. 107–117, 2004.

[12] G. Torlai, M. Guglielmo, J. Carrasquilla, M. Troyer, R. Melko, and G. Carleo, “Neural-network quantum state tomography,” *Nature Phys.*, vol. 14, no. 5, p. 447, 2018.

[13] Z. Ghahramani. “Probabilistic machine learning and artificial intelligence,” *Nature*, vol. 521, no. 7553, pp. 452–459, 2015.

[14] B. B. Sahoo, R. Jha, A. Singh, and D. Kumar, “Long short-term memory (LSTM) recurrent neural network for low-flow hydrological time series forecasting,” *Acta Geophys.*, vol. 67, no. 5, pp. 1471–1481, 2019.

[15] Y. Yu, “A review of recurrent neural networks: LSTM cells and network architectures,” *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, 2019.

[16] V. Gudivada, A. Apon, and J. Ding, “Data quality considerations for big data and machine learning: Going beyond data cleaning and transformations,” *Int. J. Adv. Software*, vol. 10, no. 1, pp. 1–20, 2017.

[17] J. Subramanian and R. Simon, “Overfitting in prediction models-is it a problem only in high dimensions?,” *Contemp. Clin. Trials*, vol. 36, no. 2, pp. 636–641, 2013.

[18] R. Reris, and J. PaulBrooks, “Principal Component Analysis and Optimization: A Tutorial,” Virginia Common wealth University Scholars Compass, USA, Vol. 212, 2015.

[19] S. Zhang, L. Lu, J. Yu, and H. Zhou, “Short-term water level prediction using different artificial intelligent models,” in: *2016 5th International Conference on Agro-Geoinformatics*, Agro-Geoinformatics 2016, 2016.

[20] S. Zainudin, D. S. Jasim, and A. A. Bakar, “Comparative analysis of data mining techniques for Malaysian rainfall prediction,” *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 6, no. 6, pp. 1148–1153, 2016.

[21] S. Chatterjee, B. Datta, S. Sen, N. Dey, and N. C. Debnathm, “Debnathm, Rainfall prediction using hybrid neural network approach,” In: *2018 2nd International Conference on Recent Advances in Signal Processing, Telecommunications & Computing* (*SigTelCom*), IEEE, 2018, pp. 67–72.

[22] J. T. Esteves, G. de Souza Rolim, and A. Sergio Ferraudo, “Rainfall prediction methodology with binary multilayer perceptron neural networks,” *Climate Dynamics*, vol. 52, no. 3–4, pp. 2319–2331, 2019.

[23] N. Tyagi and A. Kumar, “Comparative analysis of backpropagation and RBF neural network on monthly rainfall prediction,” *Proceedings of International Conference on Inventive Computation Technology* (*ICICT*) *2016*, vol. 1, 2017.

[24] C. Thirumalai, K. SriHarsha, M. Lakshmi Deepak, and K. Chaitanya Krishna. *Heuristic prediction of rainfall using machine learning techniques*, In: *2017 International Conference on Trends in Electronics and Informatics* (*ICEI*), IEEE, 2017, pp. 1114–1117.

[25] N. Solanki and G. Panchal, “A novel machine learning based approach for rainfall prediction,” In: *International Conference on Information and Communication Technology for Intelligent Systems* (*ICTIS 2017*) - *Vol. 1*, vol. 83, no. Ictis 2017, 2018.

[26] N. Mishra, H. K. Soni, S. Sharma, and A. K. Upadhyay, “Development and analysis of artificial neural network models for rainfall prediction by using time-series data,” *Int. J. Intell. Syst. Appl.*, vol. 10, no. 1, pp. 16–23, 2018.

[27] P. Singh, “Indian summer monsoon rainfall (ISMR) forecasting using time series data: A fuzzy-entropy-neuro based expert system,” *Geosci. Front.*, vol. 9, no. 4, pp. 1243–1257, 2018.

[28] J. Lei, C. Liu, and D. Jiang, “Fault diagnosis of wind turbine based on Long Short-term memory networks,” *Renew. Energy*, vol. 133, pp. 422–432, 2019.

[29] F. Kong, J. Li, and Z. Lv, “Construction of intelligent traffic information recommendation system based on long short-term memory,” *J. Comput. Sci.*, vol. 26, pp. 78–86, 2018.

[30] W. Bao, J. Yue, and Y. Rao, “A deep learning framework for financial time series using stacked autoencoders and long-short term memory,” *PloS One*, vol. 12, no. 7, 2017.

[31] B. Ay Karakuş, M. Talo, Rıza Hallaç, and G. Aydin, “Evaluating deep learning models for sentiment classification,” *Concurrency Comput. Practice Experience*, vol. 30, no. 21, p. e4783, 2018.

[32] J. Schmidhuber, and S. Hochreiter, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[33] Rainfall Dataset n.d. https://www.kaggle.com/redikod/historical-rainfall-data-in-bangladesh.

[34] B. ShenHow, and H. Loong Lam, “Sustainability evaluation for biomass supply chain synthesis: novel principal component analysis (PCA) aided optimisation approach,” *J. Cleaner Prod.*, vol. 189, pp. 941–961, 2018.

[35] Z. Lou, D. Shen, and Y. Wang, “Two-step principal component analysis for dynamic processes monitoring,” *Canadian J. Chem. Eng.*, 96, no. 1, pp. 160–170, 2018.

[36] N. Kausar, B. B. Samir, S. B. Sulaiman, I. Ahmad, and M. Hussain, “An approach towards intrusion detection using PCA feature subsets and SVM,” In: *2012 International Conference on Computer & Information Science* (*ICCIS*), vol. 2, IEEE, 2012, pp. 569–574.

[37] I. M. Kozlova, “Principal component analysis in emotion recognition: a review of the literature,” Russian Economy: Goals, Challenges and Achievements, Scientific Technologies, Russia, vol. 136, 2018.

[38] M. Kibanov, M. Becker, J. Mueller, M. Atzmueller, A. Hotho, and G. Stumme, “Adaptive kNN using expected accuracy for classification of geo-spatial data,” In: *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*, ACM, 2018, pp. 857–865.