MAUSAM, **75**, 3 (July 2024), 783-792

MAUSAM

DOI : https://doi.org/10.54302/mausam.v75i3.6239
Homepage: https://mausamjournal.imd.gov.in/index.php/MAUSAM

UDC No. 551.577 : 551.509.314 (540.49)

Hybrid deep learning algorithms on the dimensionally reduced dataset with optimized parameters for high-precision predictions of rainfall in Chhattisgarh State

NISHA THAKUR and SANJEEV KARMAKAR

Bhilai Institute of Technology, Durg, Chhattisgarh, India

(*Received 9April 2023, Accepted 23 November 2023*)

e mails : *nishathakur.india@gmail.com; dr.karmakars@gmail.com

सार– अनुक्रमिक हाइब्रिड मॉडल का उपयोग करके बहु-परिवर्त वर्षा डेटा का समय श्रृंखला पूर्वानुमान किया गया। इस मॉडल में, मूल जानकारी की न्यूनतम कमी के साथ डेटासेट के आयाम को कम करने के लिए प्रमुख घटक विश्लेषण (पीसीए) का उपयोग किया गया। डीप लर्निंग एल्गोरिदम (LSTM) में उपयोग की जाने वाली गवाक्ष आकार और लॉन्ग शॉर्ट टर्म मेमोरी (LSTM) इकाइयों की संख्या का अनुकूलित मान जेनेटिक एल्गोरिदम (GA) का उपयोग करके आकलित किया गया। 99 प्रतिशत मूल जानकारी को बनाए रखते हुए, इसके आयामों को कम करने के लिए मूल डेटासेट पर पीसीए लागू किया गया। इसके बाद, PCA का उपयोग करके प्राप्त डेटासेट को LSTM गवाक्ष आकार और इकाइयों की संख्या के अनुकूलित मान प्राप्त करने के लिए GA एल्गोरिथम में इनपुट किया गया। विभिन्न मॉडलों जैसे कि LSTM, PCA के LSTM में विलय (PCA-LSTM), GA के LSTM में विलय (GA-LSTM) तथा PCA के GA और LSTM में विलय (PCA-O-LSTM) से प्राप्त परिणामों का एक व्यापक, तुलनात्मक अध्ययन किया गया। LSTM का उपयोग पूर्वानुमान 90:10, 80:20 और 70:30 के प्रशिक्षण-परीक्षण अनुपातों के लिए किया गया, जहाँ 80:20 अनुपात ने बेहतर परिणाम प्रदान किए, इसलिए बाकी विश्लेषण के लिए 80:20 के अनुपात का उपयोग किया गया। परिणामों की बेहतर व्याख्या के लिए, प्रत्येक मॉडल को विभिन्न कालावधियों, जैसे 10, 20, 50, 100 और 200 के लिए चलाया गया। विभिन्न मॉडलों का उपयोग करके पूर्वानुमानों की गुणवत्ता का मूल्यांकन विभिन्न प्राचलों जैसे निर्धारण गुणांक (R2), माध्य वर्ग त्रुटि (MSE), मूल-माध्य-वर्ग त्रुटि (RMSE), माध्य निरपेक्ष त्रुटि (MAE), सामान्यीकृत त्रुटि (NORM), RMSE-प्रेक्षण मानक विचलन अनुपात (RSR) और कोसाइन समानता (CS) द्वारा किया गया। R2 का मान (0.962874, 0.972276), (0.970131-0.955826) और (0.950982- 0.972991) की रेंज में पाया गया, जिसमें GA-LSTM, PCA-LSTM और PCA-O-LSTM के मामले में क्रमशः 200, 200 और 100 कालावधियों के लिए उक्त प्राचलों का सर्वोत्तम मान पाया गया। R2 का सर्वोत्तम सम्भावित मान PCA-O-LSTM मॉडल के मामले में देखा गया, जिसमें GA के साथ-साथ कम आयामी डेटासेट को गवाक्ष आकार और इकाइयों की संख्या को अनुकूलित करने के लिए शामिल किया गया।

ABSTRACT. A Time series forecasting of multi-variant rainfall data was done using a sequential hybrid model. In this model, principal component analysis (PCA) was used to reduce the dimension of the dataset with minimal loss of the original information. The optimized value of window size and the number of Long Short Term Memory (LSTM) units to be used in the deep learning algorithm (LSTM) were estimated using the Genetic algorithm (GA). PCA was applied to the original dataset to reduce its dimensions, keeping 99 percent of the original information. Thereafter, the dataset retrieved using PCA was inputted to the GA algorithm to get the optimized values of LSTM window size and number of units. A comprehensive, comparative study of the results obtained from various models, such as LSTM, PCA merged to LSTM (PCA-LSTM), GA merged to LSTM (GA-LSTM), and PCA merged to GA and LSTM (PCA-O-LSTM) was carried out. The prediction using LSTM was carried out for training–testing ratios of 90:10, 80:20 and 70:30, where the 80:20 ratio provided better results therefore this ratio of 80:20 was used for the rest of the analysis. For a better interpretation of the results, each of the models was run for various epochs, like 10, 20, 50, 100 and 200. The quality of predictions using various models was evaluated by different parameters like using determination coefficient (R2), mean square error (MSE), root-mean-square error (RMSE), mean absolute error (MAE), Normalized error (NORM), RMSEobservations standard deviation ratio (RSR) and cosine similarity (CS). The value of R2 was found in the range of (0.962874, 0.972276), (0.970131-0.955826) and (0.950982- 0.972991) with the best value of the said parameter for 200, 200 and 100 epochs in case of GA-LSTM, PCA-LSTM and PCA-O-LSTM, respectively. The best possible value of R2 was seen in the case of the PCA-O-LSTM model in which a dimensional-reduced dataset along with GA optimized the window size and numbers of units were incorporated.

Key words – Prediction, Precipitation, Genetic algorithm (GA), Principal component analysis (PCA), Long-Short-Term memory network (LSTM), Optimization method.

1. Introduction

The chaotic behaviour of atmospheric conditions in applications such as rainfall has always remained one of the most dynamic parameters in nature when integrated with meteorological parameters such as minimum and maximum temperature, relative humidity pressure, and wind speed [Le *et al*., (2020); Esteves *et al*., (2019); Omar *et al.*, (2018)]. This stochastic behaviour affects many aspects of our daily lives, ranging from damage to infrastructure, both directly and indirectly Chena *et al*., (2022). It also hinders the empirical approach for forecasting, requirement of the regional crop is affected. Despite its significance in maintaining the hydrological cycle, sometimes excessive rainfall leads to flooding disasters [Barrera *et al*., (2022); Nayak *et al*., (2013)], which adversely affects societies and human civilizations.

The major challenges faced by rainfall forecasting include its stochastic and unpredictable, nonlinear behaviour. The unavailability of long-term historical data renders the rainfall forecasting process more intricate and cumbersome. With the advancement in technology, various techniques such as data mining, Support Vector Machines, artificial intelligence, fuzzy Logic, neuro-fuzzy Logic, deep learning, [Abbot and Marohasy, (2017); Davenport and Diffenbaugh (2021) and Tripathi *et al*., (2006)] and machine learning are employed in rainfall prediction. These advanced techniques can resolve the inherent stochastic and nonlinear behaviours involved in the rainfall prediction mechanism [Deo & Şahin, (2015); Hashim *et al*., (2016); Nayak *et al*., (2005); Nayak *et al*., (2004)].

The machine Learning approach has shown itself an appropriate technique for rainfall prediction by extracting the hidden patterns using historical data. Artificial Neural Networks (ANNs), Fuzzy Logic (FL), Neuro-Fuzzy Logic, Support vector Machine (SVM), random forest *etc.* [Sun *et al*., (2010); Elbeltagi *et al*., (2020),] are extensively used data mining techniques in practice. Another algorithm named the Random Forest was also reported as the most effective technique with the quality of minimized training time, with greater flexibility among the classification techniques [Breiman (2011)].

Still, high correlation in meteorological data samples is subjected to high autocorrelation, which becomes the forecasting process cumbersome with data mining approaches. Because of the lack of gradient in the network, the predictive uncertainty in rainfall forecasting was enhanced [Poornima & Pushpalatha (2019)]. To resolve this issue, deep learning approaches are becoming widely used in complex problems such as wind prediction, evaporation prediction and stream-flow prediction [Liu *et al.*, (2018); Majhi *et al*., (2020); Fu *et al*., (2020)].

In this paper, to overcome the lacuna of the Varnishing gradient, RNN-based models like Long Short Term Memory, Long Short Term Memory integrated Genetic Algorithm, Principal Component Analysis coupled Long Short Term Memory and PCA GA Long Short Term Memory are used for forecasting the time series monthly rainfall of Chhattisgarh region to reduce the errors and for getting an outmost result of prediction.

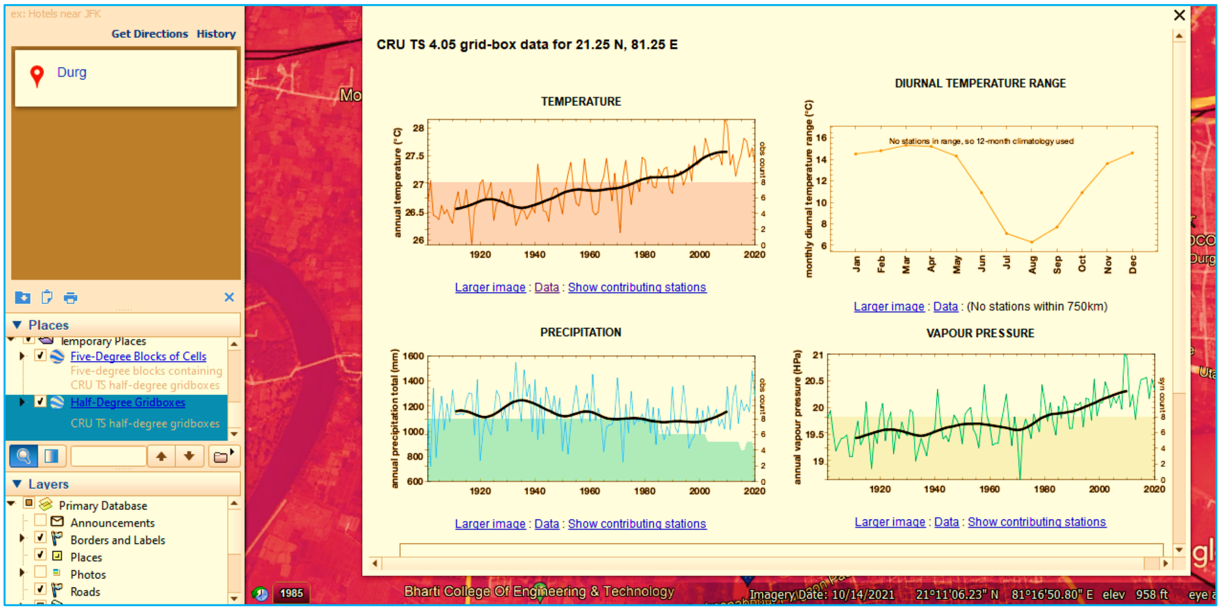
2. Dataset and methods

2.1. *Study Region*

District Durg of state Chhattisgarh is one of the most highly populated districts of the Chhattisgarh state in India. The district lies between 20° 54′ and 21° 32′ north latitude& 81° 10′ and 81° 36′ east longitude. The climate of the district is tropical type. Summer is a little hotter. The temperature rise begins from March to May. May is the hottest among others. Durg district's annual average rainfall is 1052 mm. During the year, most rainfall occurs during the monsoon months, from June to September. July is the month of the highest rainfall available at the site given below:-
[https://durg.gov.in/aboutdistrict/#:~:text=Durg%20district%20is%20situated%20in,81%C2%B036%E2%80%B2%20east%20longitude]

2.2. *Dataset*

Google Earth Interface “CRU TS Version 4.05” was downloaded from the website of the Climate Research Unit (CRU). CRU is a section of the University of East Anglia where one may avail a worldwide historical record of over-land temperature data. This information is available as a cover on Google Earth. CRU TS Version 4.05 was opened in Google Earth Pro and Durg, Chhattisgarh, India was selected to retrieve the historical climate information of the location. The data was



CRU TS 4.05 grid-box data for 21.25 N, 81.25 E

Larger image : Data : Show contributing stations

Larger image : Data : (No stations within 750km)

No stations in range, so 12-month climatology used

Fig. 1. Downloading process of historic climate data using Google Earth Interface "CRU TS Version 4.05"

associated withgrid-box data 21.25° N, 81.25° E Temperature (°C), Diurnal Temperature range (°C), Precipitation (mm/month) and Vapour Pressure (hPa) were the four parameters under which the monthly data from 1901 to 2020 was downloaded. A screenshot of the same is provided in Fig. 1.

2.3. *Methods adopted*

With a target to compare and find the best possible way for rainfall prediction, the LSTM technique was integrated into Principal Component Analysis (PCA) and Genetic Algorithm (GA).

2.3.1. *Genetic algorithm Integrated into Long Short-Term Memory (GA-LSTM)*

In this study, a hybrid approach of the LSTM network coupled with GA for finding the time window and number of LSTM units for rainfall time series forecasting. The functioning of LSTM has already been discussed in previous work [Goodfellow *et al*., (2016); Schmidhuber and Hochreiter, (1997); Gers *et al*., (1999); Kim *et al*., (2017); Armano *et al*., (2005)]. In the evolutionary search algorithm, GA was used to identify the optimal size of time windows and the number of LSTM units. To provide higher order of stability to the model concerned, previous time step is included to the input given to the model. For example, an input window of size 16 will include our current time-step along with the 15 previous time-steps. It has been reported by many of the researchers that selecting the proper size of a sliding window, leads to the improved efficiency of the model [Tomer *et al*. (2022)]. The minimum value of Root-mean-square error (RMSE) was considered to be the fitness function while optimizing the value of window size and the number of LSTM Units. The detailed process of GA-LSTM is depicted in Fig. 2(a) [Chung and Shin, (2018); Armano *et al*., (2005); Kim and Shin, (2007)].

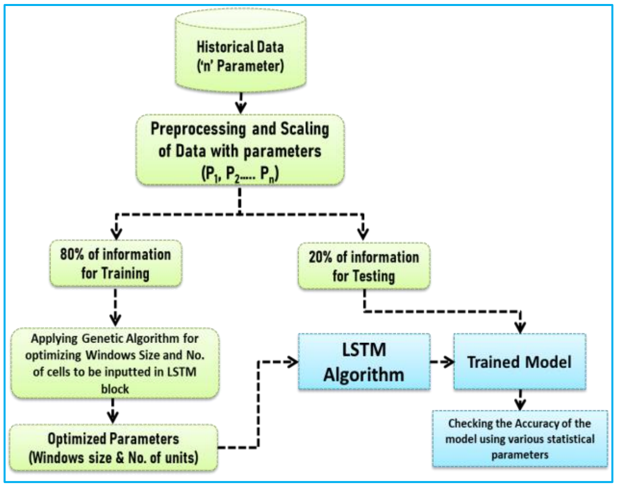

Historical Data
('n' Parameter)
Preprocessing and Scaling of Data with parameters
(P_1, P_2..... P_n)
80% of information for Training
20% of information for Testing
Applying Genetic Algorithm for optimizing Windows Size and No. of cells to be inputted in LSTM block
Optimized Parameters
(Windows size & No. of units)
LSTM Algorithm
Trained Model
Checking the Accuracy of the model using various statistical parameters

Fig. 2(a). Flowchart of the algorithm (GA-LSTM) used

A stepwise description of the GA-LSTM algorithm used is depicted below:

(*i*) Historical monthly rainfall data with an 'n' independent parameter was inputted.

(*ii*) Inputted data was preprocessed and scaled before further implementation.

(*iii*) Preprocessed data was split in 80:20 for training and testing purposes, respectively.

(*iv*) A genetic Algorithm (GA) is applied to find the optimum values of window size and the number of LSTM units.

(*v*) Optimum values of windows size and the number of LSTM units got from step (04) were used to run the LSTM algorithm to create a trained model.

(*vi*) The 20% of overall data, which was kept for testing purposes, was used to compare the predicted and actual rainfall data.

(*vii*) The accuracy of prediction was measured through various parameters like Root mean square error (RMSE), Cosine Similarities (CS) *etc*.

The above-mentioned algorithm (GA-LSTM) may also be represented mathematically as below:-

$$W_s = (\alpha_i \in R : i \in R)$$
$$U = (\beta_i \in R : i \in R)$$
$$D = (\gamma_i \in R : i \in R)$$
$$S_{\text{test}} = (\exists x_i : x_i \in D : i = 1,2,3,\ldots n*0.2)$$
$$S_{\text{tr}} = (\exists x_i : x_i \in D : i = 1,2,3,\ldots n*0.8)$$
$$S_{\text{test}} \cap S_{\text{Str}} = \phi$$
$$\text{RMSE}_{x-1} = \text{predict}_{\text{LSTM}}(W_s, U, Str_{x-1}) - S_{\text{test}}$$
$$\text{Str}_{x-1} = \text{Str}_x$$
$$\text{RMSE}_{x-1} = \text{RMSE}[0]$$

```
do
{
$(\text{Str}_x = \text{RMSE}[0]) =$
$\text{GA}\{\text{Unicros}[Str_x - 1, W_s(x-1), U_n(x-1), \text{Bitwise}$
$[Str_x - 1, W_s(x-1), U_n(x-1), U_n(x-1)]]\}(Str_x - S_{test})$
$x = x - 1$
}
while (num generation ==n)
location = $x$
for$i$ = 0 to $i<x$
{
if [RMSE ($i$) < RMSE (location)]
location = $i$
$i = i + 1$}
RMSE = Predict$_{\text{LSTM}}$ [W$_s$ (location), Un (location),
Str$_{\text{location}}$ – S$_{\text{test}}$)
```

Algorithm for implementation of GA-LSTM

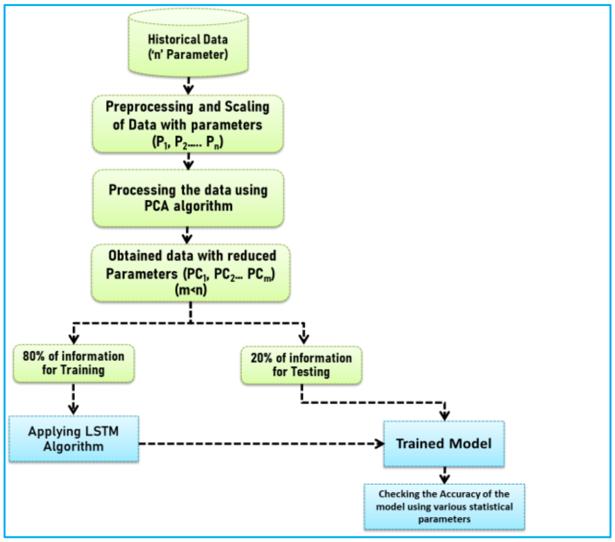

Historical Data
('n' Parameter)
Preprocessing and Scaling of Data with parameters
(P1, P2..... Pn)
Processing the data using PCA algorithm
Obtained data with reduced Parameters (PC1, PC2... PCm)
(m<n)
80% of information for Training
20% of information for Testing
Applying LSTM Algorithm
Trained Model
Checking the Accuracy of the model using various statistical parameters

Fig. 2(b). Flowchart of the algorithm (PCA-LSTM) used

2.3.2. *Principal Component Analysis coupled Long Short term memory (PCA-LSTM)*

The Principal Component Analysis (PCA) method put efforts to reduce the dimensionality of the data set with a minimum loss of original information and extract characteristics in various fields such as image, voice and other data problems. For classifying and prediction in data processing, dimension reduction is a very important component. There can be many independent parameters or features in the dataset considered, which is not of so much importance so far as the prediction of the dependent parameter is concerned. There is no point in considering the said parameter and increasing the complexity of the process of prediction. It could be noted that the dimensionally reduced dataset makes the analysis much convenient and machine learning process without incompatible parameters to process. Having a non-required independent parameter may increase the possibilities of over-learning or overfitting while training the prediction model. Therefore, reducing the dimension can be an excellent technique to be adopted [Kim and Shin (2007), Hou (2021)]. In present case a multivariate dataset of 4 variable were used for analysis. PCA was applied to the said dataset of 4 parameters, which reduced the dataset into single variable containing the 99% of information carried by the original dataset. In this way, smaller dataset was inputted without much loss of its originality.

For completion of the task to predict the monthly rainfall using a hybrid algorithm PCA-LSTM, in which dimensionally reduced information was to be inputted in the LSTM section is depicted in form of a flowchart. The flowchart is provided in Fig. 2(b).

A stepwise description of the PCA-LSTM algorithm used is depicted below:

(*i*) Historical monthly rainfall data with an 'n' independent parameter was inputted.

(*ii*) Inputted data was preprocessed and scaled before further implementation.

(*iii*) Principal Component Analysis (PCA) algorithm was applied to the scaled data retrieved from step (2) To reduce the number of independent parameters (m) without minimizing the significance of the data to a large extent. It is expected that the reduced data using PCA minimizes the complexity of the information. This is to note that m<n.

(*iv*) Data with reduced independent parameters (m) was split in 80:20 for training and testing purposes, respectively.

(*v*) Training data was used to run the LSTM algorithm to create a trained model.

(*vi*) The 20% of overall data, which was kept for testing purposes, was used to compare the predicted and actual rainfall data. The accuracy of prediction was measured through various parameters like Determination Coefficient (R^2), Mean square error (MSE), Root-mean-square error (RMSE), Mean absolute error (MAE), Normalized error (NORM), Cosine Similarities (CS) *etc.*

The above-mentioned algorithm (PCA-LSTM) may also be represented mathematically as below:-

$X = \{x_1, x_2, \ldots x_N\}$

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$\mathrm{Cov}(x) = \frac{1}{N}\sum_{i=1}^{N} (x_i - \bar{x}_i)(x_i - \bar{x}_i)^T$$

$\gamma_1, \gamma_2, \ldots \gamma_D$ (Eigen Vectors)

$\lambda_1, \lambda_2, \ldots \lambda_D$ (Corresponding Eigen Values)

$\lambda_1 \geq \lambda_2 \geq, \ldots \geq \lambda_D$ (Arranging in decreasing order)

$$\gamma = \left[\gamma_1^T(x-\bar{x}), \gamma_2^T(x-\bar{x}), \ldots (x-\bar{x})\right] \in \mathrm{R}^{\mathrm{D}}$$

(where $d \leq D^*$) (New Lower Dimension Matrix)

$$D \approx \bar{x} + \left[\gamma_1^T(x-\bar{x})\gamma_1, \gamma_2^T(x-\bar{x})\gamma_2, \ldots \gamma_d^T(x-\bar{x})\gamma_d\right]$$

(Approximation of original data)

$$S_{tr} = \{\exists x_i : y \in D : i = 1,2,3..n*0.8\}$$

$$S_{test} = \{\exists x_i : y \in D : i = 1,2,3..n*0.2\}$$

$$S_{test} \cap S_{Str} = \phi$$

RMSE = LSTM (S_{tr} – S_{test}) (Finding RMSE for predicted information)

Algorithm for implementation of PCA-LSTM

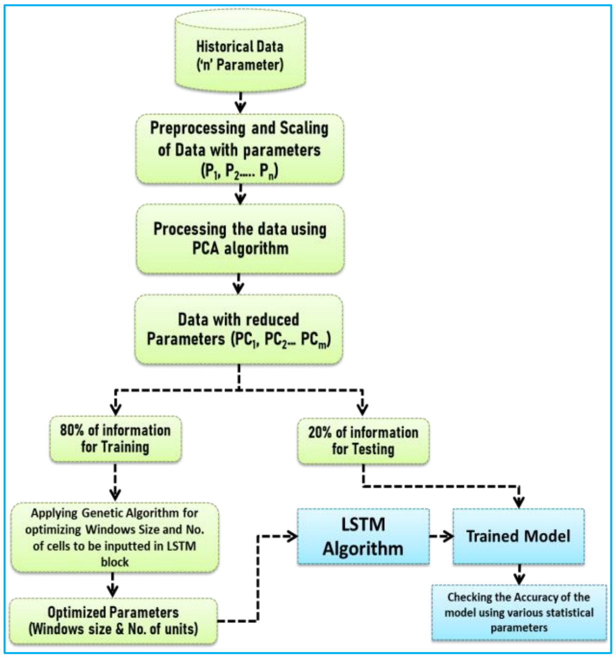

Historical Data ('n' Parameter)
Preprocessing and Scaling of Data with parameters (P_1, P_2..... P_n)
Processing the data using PCA algorithm
Data with reduced Parameters (PC_1, PC_2... PC_m)
80% of information for Training
20% of information for Testing
Applying Genetic Algorithm for optimizing Windows Size and No. of cells to be inputted in LSTM block
Optimized Parameters (Windows size & No. of units)
LSTM Algorithm
Trained Model
Checking the Accuracy of the model using various statistical parameters

Fig. 2(c). Flowchart of the algorithm (PCA-O- LSTM) used

2.3.3. *Principal Component Analysis coupled Genetic Algorithm integrated with Long Short-Term Memory (PCA-O-LSTM)*

Principal component analysis (PCA) minimizes the complexity of the dataset by transforming it into reduced dimensions without losing significant information [Hou,(2021)].Genetic Algorithm allows us to find suitable parameters for performing prediction. In this section, an approach of prediction implementing PCA and then GA subsequently is discussed. PCA was used to minimize the dimensionality and optimized values of window size and numbers of LSTM units were determined using GA. The detailed algorithm in form of a flow chart is provided in Fig. 2(c).

A stepwise description of the PCA-O- LSTM algorithm used is depicted below:

(*i*) Historical monthly rainfall data with an 'n' independent parameter was inputted.

(*ii*) Inputted data was preprocessed and scaled before further implementation.

(*iii*) Principal Component Analysis (PCA) algorithm was applied to the scaled data retrieved from step (2) to reduce the number of independent parameters (m) without minimizing the significance of the data to a large extent. It is expected that the reduced data using PCA minimizes the complexity of the information. This is to note that m<n.

(*iv*) Data with reduced independent parameters (m) was split in 80:20 for training and testing purposes, respectively.

(*v*) A genetic Algorithm (GA) is applied to find the optimum values of window size and the number of LSTM units.

(*vi*) Optimum values of windows size and the number of LSTM units got from step (05) were used to run the LSTM algorithm to create a trained model.

(*vii*) The 20% of overall data, which was kept for testing purposes, was used to compare the predicted and actual rainfall data.

(*viii*) The accuracy of prediction was measured through various parameters like Root mean square error (RMSE), Cosine Similarities (CS) *etc*.

The above-mentioned algorithm (PCA-O-LSTM) may also be represented mathematically as below:-

$X = \{x_1, x_2, \ldots x_N\}$

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$\mathrm{Cov}(x) = \frac{1}{N}\sum_{i=1}^{N} (x_i - \bar{x}_i)(x_i - \bar{x}_i)^T$$

$\gamma_1, \gamma_2, \ldots \gamma_D$ (Eigen Vectors)

$\lambda_1, \lambda_2, \ldots \lambda_D$(Corresponding Eigen Values)

$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_D$(Arranging in decreasing order)

$\gamma = \left[\gamma_1^T(x-\bar{x}), \gamma_2^T(x-\bar{x}), \ldots \gamma_d^T(x-\bar{x})\right] \in \mathrm{R}^{\mathrm{D}}$ (where d $\leq D$*) (New Lower Dimension Matrix)

$D \approx \bar{x} + \left[\gamma_1^T(x-\bar{x}), \gamma_1, \gamma_2^T(x-\bar{x})\gamma_2, \ldots \gamma_d^T(x-\bar{x})\right] \gamma_d$ (Approximation of original data)

$$S_{tr} = \{\exists x_i : y \in D : i = 1,2,3 \ldots n * 0.8\}$$

$$S_{test} = \{\exists x_i : y \in D : i = 1,2,3 \ldots n * 0.2\}$$

$$S_{test} \cap S_{Str} = \phi$$

```
RMSE$_{x-1}$ = predict$_{\mathrm{LSTM}}$ ($W_s$, U, Str$_{x-1}$) – Str$_{\mathrm{test}}$)
Str$_{x-1}$ –Str$_x$
RMSE$_{x-1}$ = RMSE [0]
Do
{
(Str$_x$, RMSE [0])
GA (Unicros (Str$_x$ – 1, $W_s$[$x$–1]), $U_n$[$x$–1], Bitwise (Str$_x$ – 1, $W_s$[$x$–1]), $U_n$[$x$–1]), $U_n$[$x$–1]) (Str$_x$ – $S_{\mathrm{test}}$)
$x$ = $x$ – 1
}
While (numgeneration = = $n$)
location = $x$
for$i$ = 0 to $i$<$x$
{
If (RMSE [i] < RMSE [location])
location = $i$
$i$ = $i$ + 1
}
RMSE = Predict$_{\mathrm{LSTM}}$ (W$_s$[location], Un[location], Str$_{\mathrm{location}}$ – $S_{\mathrm{test}}$)
```

Algorithm for implementation of PCA-O-LSTM

3. Results and discussion

In this study, surface meteorological rainfall parametric data from 1901 to 2020 of Durg (Chhattisgarh, India) was considered. Various experiments were performed to get a higher accuracy rainfall forecast model.

It is well known that the LSTM models had a significant capacity for forecasting precipitation. The necessity of researching merged models like GA-LSTM, PCA-LSTM and PCA-O-LSTM, may enhance the prediction outcomes at particular periods. The evaluation of discussed hybrid models and their comparison with the standalone LSTM model has been done.

Three different ratios for training and testing data, *i.e.*, 90:10, 80:20 and 70:30 respectively, were considered for predicting rainfall using Long Short Term Memory (LSTM) algorithm. In each of the ratios, 10, 20, 50, 100 and 200 epochs along with fixed window size (30) and the number of units (64) were used. This is done to identify the best possible ratio to study the rest of the merged algorithms reported in this article.

The excellence of the proposed models was estimated in terms of R^2, MSE, RMSE, MAE, NORM, RSR and CS. By looking at Table 1, it may be concluded that the splitting ratio of 80:20 improves the prediction results compared to the 90:10 and 70:30 ratios. It may further bereferred to the fact that the value of R^2 was found to be in the range (0.6844-0.9648), (0.9002-0.9754), (0.2968-0.9528) for 90:10, 80:20 and 70:30 training testing data, respectively. Therefore, the training and validation data set was split into an 80:20 ratio for the rest of the analysis, considering the different algorithms reported.

In case, if the training data size is sufficiently large, 80% of the total data is enough for proper training but when the data size is not sufficiently large, 90% of the total data for training purposes may be used for obtaining good prediction results. Taking 90% of a huge data set may be a cause of overlearning, resulting in confusion or improper prediction. However, using 70% of the total dataset for training purposes may decrease the training standard, consequently less accurate prediction [available

TABLE 1

Comparative Evaluation table between the models used

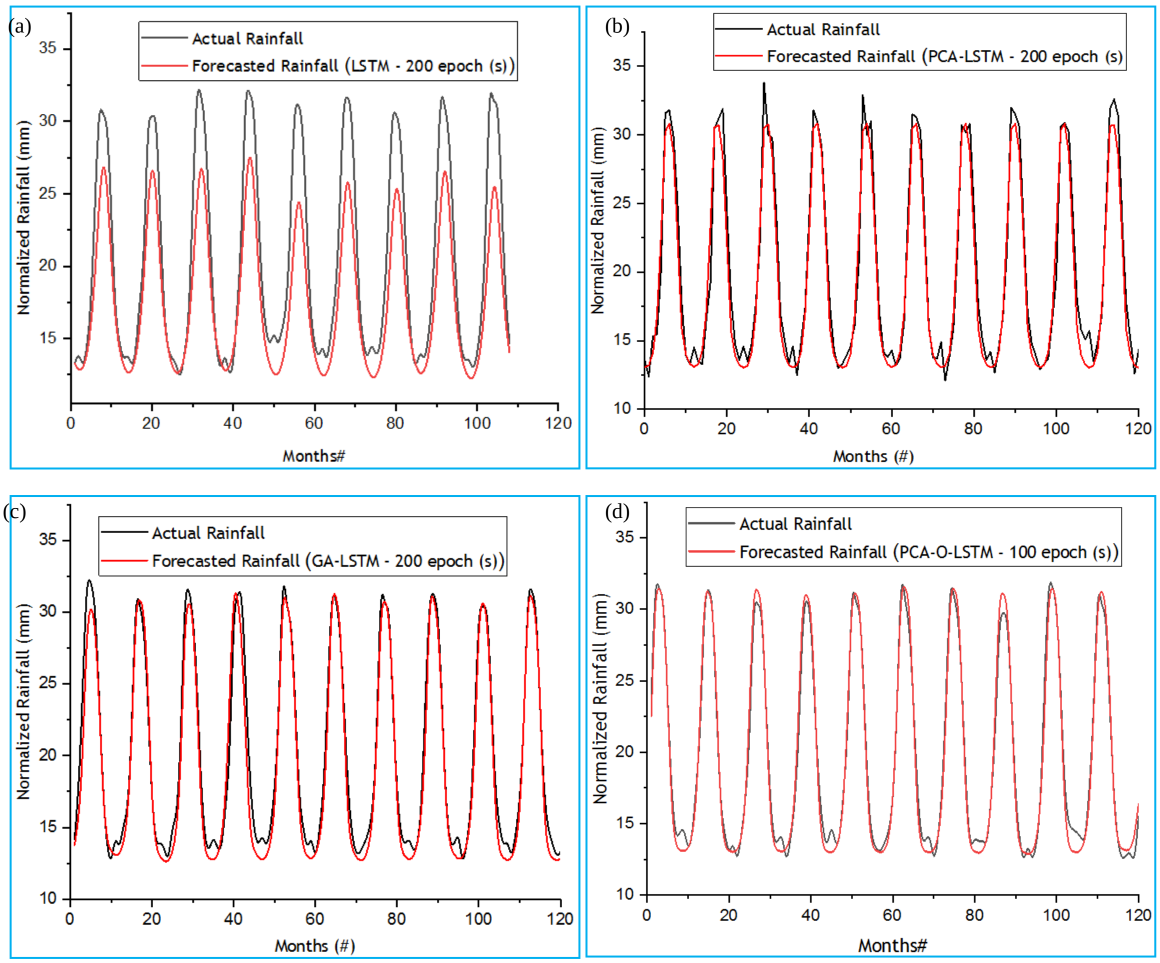
S. No.	Algorithm	epoch (s)	Train: Test	Window Size	No. of Units	R^2	MSE	RMSE	MAE	NORM	RSR	CS
1	LSTM	10	90:10	30	64	**0.684409**	16.13297	4.016587	2.765619	80.63242	0.561775	0.984931
2		20	90:10			**0.939859**	3.0761	1.753881	1.319111	35.20892	0.245237	0.997456
3		50	90:10			**0.930809**	3.537396	1.880797	1.397861	37.75673	0.263041	0.997654
4		100	90:10			**0.958748**	2.107213	1.451624	1.166353	29.14115	0.203107	0.998451
5		200	90:10			**0.964887**	1.792843	1.338971	1.04223	26.87965	0.187384	0.998554
6	LSTM	10	80:20	30	64	0.900232	4.901013	2.213823	1.66882	45.26172	0.315861	0.995934
7		20	80:20			0.937298	3.080184	1.755045	1.291112	35.88199	0.250404	0.997211
8		50	80:20			0.951966	2.359622	1.536106	1.150963	31.40576	0.219166	0.997749
9		100	80:20			0.972715	1.397757	1.182268	0.925069	23.73386	0.165182	0.998927
10		200	80:20			0.975487	1.246168	1.116319	0.891312	22.40995	0.156566	0.998927
11	LSTM	10	70:30	30	64	0.296868	33.69989	5.805161	4.335277	120.2383	0.83853	0.962465
12		20	70:30			0.89762	4.919565	2.218009	1.480465	45.9401	0.319969	0.995397
13		50	70:30			0.841896	7.54291	2.746436	1.849259	56.88504	0.397623	0.994486
14		100	70:30			0.896125	4.99348	2.23461	1.461487	46.28394	0.322296	0.99544
15		200	70:30			0.952867	2.315357	1.52163	1.150622	31.10979	0.217101	0.997879
16	GA-LSTM	10	80:20	36	6	**0.962874**	1.901884	1.379088	1.056323	27.685	0.192681	0.998532
17		20	80:20	13	6	**0.965413**	1.771822	1.331098	1.012147	26.72161	0.185976	0.99857
18		50	80:20	34	4	**0.970695**	1.501239	1.225251	0.95519	24.59673	0.171187	0.998797
19		100	80:20	36	2	**0.972276**	1.414454	1.189308	0.95337	23.8752	0.166507	0.998911
20		200	80:20	37	7	**0.972276**	1.414454	1.189308	0.95337	23.8752	0.166507	0.998911
31	PCA-LSTM	10	80:20	30	64	**0.955826**	2.253698	1.501232	1.224522	30.13702	0.210177	0.997953
32		20	80:20			**0.956674**	2.21043	1.486751	1.206417	29.84633	0.208149	0.998079
33		50	80:20			**0.952905**	**2.313482**	**1.521014**	**1.13831**	**31.0972**	**0.217013**	**0.998041**
34		100	80:20			**0.908629**	4.385184	2.094083	1.425011	43.37331	0.302276	0.99559
35		200	80:20			**0.970131**	**1.523862**	**1.234448**	**1.000287**	**24.78137**	**0.172826**	**0.99874**
46	PCA-O-LSTM	10	80:20	28	2	**0.950982**	2.491911	1.578579	1.269686	31.68975	0.221399	0.997298
47		20	80:20	19	10	**0.946010**	2.744698	1.656713	1.303754	33.25828	0.232358	0.997714
48		50	80:20	8	6	**0.955898**	2.24199	1.497328	1.202976	30.05864	0.210004	0.997901
49		**100**	**80:20**	**16**	**6**	**0.972991**	**1.373043**	**1.171769**	**0.93874**	**23.5231**	**0.164343**	**0.998657**
50		200	80:20	60	2	**0.972715**	1.397757	1.182268	0.925069	23.73386	0.165182	0.998705

at website https://vitalflux.com/machine-learning-training-validation-test-data (accessed on 1/4/2022)].

To prove the ascendancy of the proposed model (s), the prediction performance (s) retrieved from LSTM, GA-LSTM, PCA-LSTM and PCA-O-LSTM were compared through the same parameter used earlier, *i.e.*, R^2, MSE, RMSE, MAE, NORM, RSR and CS.

The LSTM, GA-LSTM, PCA-LSTM and PCA-O-LSTM-based models were run for 5, 10, 20, 50, 100 and 200 epochs respectively using 80% of the total dataset all the time. On average, it took over 200 epochs for each rainfall data to reach minimum error and high accuracy for forecasting. The factors MSE, RMSE, MAE, NORM and RSR need to be less for obtaining optimum prediction quality. The values of Cosine similarity (CS) and Correlation Coefficient (R^2) must be close to one for the highest possible accuracy. When these values are close to 1, we consider the actual and predicted data are very similar and when the said values are close to 1, we may conclude about the usefulness of the prediction made.

If we observe Table 1 and compare all the three proposed merged models by looking at one of the parameters to judge the accuracy of model R^2, the values



Figs. 3(a&b). Comparison between Actual & Forecast Rainfall, (a) [200 epoch (s) window size=30, number of LSTM units=64] using LSTM algorithm, (b) [200 epoch (s) optimal window size=30, number of LSTM units=64] using PCA-LSTM algorithm, (c) [200 epoch (s) optimal window size=37, number of LSTM units=7) using GA-LSTM algorithm and (d)[100 epoch (s) optimal window size=16, number of LSTM units=6) using PCA-O-LSTM algorithm

are in the range (0.962874, 0.972276), (0.955826-0.970131) and (0.950982- 0.972991) with the best value of the said parameter for 200, 200 and 100 epochs in case of GA-LSTM, PCA-LSTM and PCA-O-LSTM, respectively. These results are indicative that the best possible value of R^2 was seen in the case of the PCA-O-LSTM model, in which a dimensional reduced dataset along with GA optimized the window size and numbers of units were incorporated. It is observed in the above said Table 1 that the optimized value of window size and the number of units were different for various epochs in GA-LSTM and PCA-O-LSTM algorithm results, whereas a fixed value of the same, *i.e.*, 30 and 60 respectively was used for PCA-LSTM algorithm. It is also noticeable that higher epochs produced better results. For every case except, the PCA-O-LSTM model 100 epochs gave the best prediction results. We retrieved the best prediction results for 100 epochs in the PCA-O-LSTM model.

TABLE 2

Definition of Classes

Range of Raw data	Class Number assigned
Less than 0	1
0-4	2
5-8	3
9-12	4
13-16	5
17-20	6
21-24	7
25-28	8
29-32	9
33-36	10

TABLE 3

Confusion Matrix Parameters

Class Value	n (truth)	n (classified)	Accuracy	Precision	Recall	F1 Score
4	0	3	99.25%	0	0	0
5	201	170	89.83%	0.97	0.82	0.89
6	32	63	89.83%	0.43	0.84	0.57
7	66	43	94%	1	0.65	0.79
8	2	22	95%	0.091	1	0.17
9	98	84	95.53%	0.98	0.84	0.9
10	4	18	95.53%	0.11	0.5	0.18

Figs. 3(a-d) provides a comparative view of actual and predicted data using LSTM, PCA-LSTM, GA-LSTM and PCA-O-LSTM algorithms, respectively. In these figures, a great resemblance between the actual and forecast data can be observed. The black solid line shows the actual rainfall and the red solid line show the predicted rainfall. Even from these figures, it may be observed that PCA-O-LSTM provided the most accurate results with a high similarity between the actual and predicted data.

Further one more analysis has been done for accuracy by plotting the confusion matrix the raw test data and prediction data were converted in various classes. The criterion used for classification of the raw data is given below:

The prediction done using PCA-O-LSTM algorithm at 100 epoch (s) with optimal window size = 16 and optimum number of LSTM units = 6.

Overall True Positive (TP) values found was 321 out of 403 data whereas overall accuracy of prediction was found to be 79.65 for the classification illustrated in Table 2.

Table 3 provided the detailed information about Accuracy, Precision, Recall and F1 Score for each of the classes. The table also elaborates that the Testing data contains 0, 201, 32, 66, 2, 98 and 4 whereas the Prediction data contains 3, 170, 63, 43, 22, 84 and 18 frequency of the data of the class 4, 5, 6, 7, 8, 9, 10 respectively. The same occurrence may also be noted in the Confusion Matrix provided in Fig. 4. Maximum frequency can be noted in Class value 05 and the accuracy and recall value for the same is noted as 0.97 and 0.82 respectively.

It is understandable that the values of the Confusion Matrix parameters may also be reported as more accurate, if the class interval is larger. In such cases, the values reported nearby to the diagonal of the matrix may be merged to the diagonal itself so the True Positive (TP) and False Negative (FN) cases can be increased and the model used may look a better one.

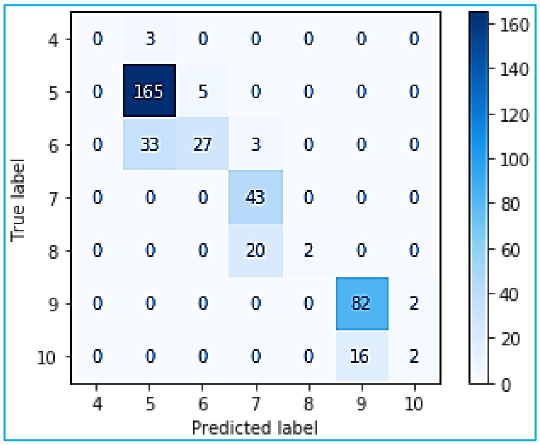


Fig. 4. Confusion Matrix for the data illustrated in Fig. 3(d)

4. Conclusions

In this paper, an innovative PCA + optimized GA-LSTM algorithm was proposed for rainfall forecasting based on the multi-variant time series dataset. Evaluation and analysis have been done using PYTHON v3.9. An attempt had been made to forecast the monthly rainfall of 1428 months by applying the LSTM algorithm using 10, 20, 50, 100 and 200 epochs. The experimental factors R^2, MSE, RMSE, MAE, NORM, RSR and CS were calculated to judge the quality of forecasting. The three proposed merged models were compared along with the standalone model to judge the accuracy of the models considered. The value of R^2, were in the range (0.900232, 0.975487), (0.962874, 0.972276), (0.955826-0.952905) and (0.950982- 0.972991) with the best value of the said parameter for 200, 200, 200 and 100 epochs in case of LSTM, GA-LSTM, PCA-LSTM and PCA-O-LSTM, respectively. These results are indicative that the best possible value of R^2 was seen in the case of the PCA-O-LSTM model, in which a dimensional reduced dataset along with GA optimized the window size and numbers of units were incorporated.

In a conclusion, Results got revealed that the proposed PCA-O-LSTM model outperforms achieving the lowest value of RMSE, MSE, MAE, NORM and RSR in comparison with the stand-alone LSTM model except the case of 200 epochs where the results obtained through LSTM model was found slightly better. Likewise, the ranks given according to performance evaluation, are PCA-O-LSTM > GA-LSTM > PCA-LSTM>LSTM.

Disclaimer: The contents and views expressed in this research paper/article are the views of the authors and do not necessarily reflect the views of the organizations they belong to.

References

Abbot and Marohasy, 2017, "The Application of Machine Learning for Evaluating Anthropogenic versus Natural Climate Change", *Geo. Res. J.*, **14**, 36-46.

Armano. G., Marchesi. M. and Murru 2005, "Hybrid genetic-neural architecture for stock indexes forecasting", *Information sciences*, **170**, 3-33.

Barrera, Y. A., Oyedele, O. L., Bilal, M., and Akinosho, T., 2022, "Rainfall prediction: A comparative analysis of modern machine learning algorithms for time-series forecasting", *Machine Learning and Applications*, **7**, 1-20.

Breiman, 2011, "Random forests", *Machine Learning*, **45**, 1, 5-32, https://doi.org/10.1023/A:1010933404324.

Chung and Shin, 2018, "Genetic Algorithm-Optimized Long Short-Term Memory Network for Stock Market Prediction", *Sustainability*,**10**, 3765.doi:10.3390/su10103765.

Chen. C., Zang. Q., Kashani. H. M., Jun. C., Bateni. M. S. and Band. S. S ,2022, "Forecast of rainfall distribution based on fixed sliding window long short-term memory", *Engineering Applications of Computational Fluid Mechanics*, **16**, 1, 248-261.

Davenport and Diffenbaugh, 2021, "Using Machine Learning to Analyze Physical Causes of Climate Change: A Case Study of U.S. Midwest Extreme Precipitation", *Geophysical Research Letters*, https://doi.org/10.1029/2021GL093787.

Deo and Şahin, 2015, "Application of the extreme learning machine algorithm for the prediction of monthly Effective Drought Index in eastern Australia", *Atmospheric Research*, **153**, 512-525.

Esteves. T.J., Rolim. S. D. and Ferraudo. S. A., 2019, "Rainfall prediction methodology with binary multilayer perceptron neural networks", *Climate Dynamics*,**52**, 3-4, 2319-2331.

Elbeltagi, A., Deng, J., Wang, K., Malik. A., 2020, "Modeling long-term dynamics of crop evapotranspiration using dep learning in a semi-arid environment", *Agricultural Water Management*, https://doi.org/10.1016/j.agwat.2020.10334.

Fu. M., Fan. T., Ding. Z. and Salih. S., 2020, "Deep learning data-intelligence model based on adjusted forecasting window scale: Application in daily streamflow simulation", *IEEE Access*, **8**, 32632-32651, https://doi.org/10.1109/ACCESS.2020.2974406.

Goodfellow. I., Bengio. Y., Courville. A. and Bengio. Y., 2016, Deep Learning; MIT Press: Cambridge, MA, USA, 373-418.

Gers. F., Schmidhuber. J. and Cummins. F., 1999, "Learning to forget: Continual prediction with LSTM", *Neural Computation*, **12**, 2451-2471.

Hou, 2021, Principal Component Analysis and Prediction of Students' Physical Health Standard Test Results Based on Recurrent Convolutio Neural Network, Hindawi Wireless Communications and Mobile Computing Volume 2021, Article ID 2438656, 11 pages https://doi.org/10.1155/2021/2438656.

Hashim, R., Roy, C., Motamedi, C., Shamsirband. S., 2016, "Selection of meteorological parameters affecting rainfall estimation using neuro-fuzzy computing methodology", *Atmospheric Research*, **171**,21-30.

https://durg.gov.in/aboutdistrict/#:~:text=Durg%20district%20is%20situated%20in,81%C2%B036%E2%80%B2%20east%20longitude. (accessed on 31/3/2022).

https://vitalflux.com/machine-learning-training-validation-test-data (accessed on 1/4/2022).

Kim. Y., Roh. J. H. and Kim H., 2017, "Early Forecasting of Rice Blast Disease Using Long Short-Term Memory Recurrent Neural Networks", *Sustainability*, **10**, 34-53.

Kim and Shin, 2007, "A hybrid approach based on neural networks and genetic algorithms for detecting temporal patterns in stock markets", *Applied Soft Computing*, **7**, 569-576.

Liu. H., Mi. X. and Li. Y., 2018, "Wind speed forecasting method based on deep learning strategy using empirical wavelet transform, long short term memory neural network and Elman neural network", *Energy Conversion and Management*, **156**, 498-514.

Le. T.T., Pham. T., Ly. H. and Shirzadi. A., 2020, "Development of 48-hour Precipitation Forecasting Model using Nonlinear Autoregressive Neural Network, Innovation for Sustainable Infrastructure", *Lecture Notes in Civil Engineering*, **54**. Springer, Singapore.

Nayak. R. D., Mahapatra. A. and Mishra. P., 2013, "Survey on rainfall prediction using artificial neural network", *International Journal of Computer Applications*, **72**, 16, 32-40. https://doi.org/10.5120/12580-9217.

Nayak, C. P., Sudheer, P. K., Ramashastri, S. K., 2005, "Fuzzy computing based rainfall–runoff model for real time flood forecasting", *Hydrological Processes*, **19**, 955-968.

Nayak, C. P., Sudheer, P. K., Ramashastri, S. K., 2004, "A neuro-fuzzy computing technique for modeling hydrological time series", *Journal of Hydrology*, **291**, 52-66.

Majhi. B., Naidu. D., Mishra. P. A. and Satapathy, C. S., 2020, "Improved prediction of daily pan evaporation using deep-LSTM model", *Neural Computing and Applications*, **32**, 12, 7823-7838. https://doi.org/10.1007/s00521-019-04127-7.

Omar. A., Shatnawiet. N. and Matouq. M., 2018,"Nonlinear Multivariate Rainfall Prediction in Jordan Using NARX-ANN Model with GIS Techniques", *Jordan Journal of Civil Engineering*, **12**, 3.

Poornima and Pushpalatha, 2019, "Prediction of rainfall using intensified LSTM based recurrent neural network with weighted linear units", *Atmosphere*, **10**, 11, p668.

Sun, Y., Zhou, Q., Xie, X., and Liu, R., 2010, "Application of the deep learning for the prediction of rainfall in Southern Taiwan, In Artificial Intelligence and Computational Intelligence (AICI)", International Conference on, 252-256.

Schmidhuber and Hochreiter, 1997, "Long short-term memory", *Neural Computation*, **9**, 8, 1735-1780.

Tomar, D., Tomar, P., Bhardwaj, A. and Sinha, G. R., 2022, "Deep Learning Neural Network Prediction System Enhanced with Best Window Size in Sliding Window Algorithm for Predicting Domestic Power Consumption in a Residential Building", *Computational Intelligence and Neuroscience*, https://doi.org/10.1155/2022/7216959.

Tripathi, S. Srinivas V. V., and Nanjundiah, S. R., 2006, "Downscaling of Precipitation for Climate Change Scenarios: A support vector machine approach", *Journal of Hydrology*, **330**, 3-4, 621-640.3.

Thakur, N., Karmakar, S., and Soni, S., 2022, "Time Series Forecasting for Uni-Variant data using optimized Long Short Term Memory Integrated with Genetic Algorithm and Performance Evaluations", *International Journal of Information Technology*,**14**, 04, 1961-1966.

Thakur. N., and Karmakar. S., 2023, Hybrid deep learning algorithms for forecasting air quality index using dimension reduction technique in search of precise results, *International Journal of Information Technology*, **15**, 04, 3181-3187.