

Deep learning model for daily rainfall prediction: case study of Jimma, Ethiopia

Demeke Endalie ^{a,*}, Getamesay Haile^a and Wondmagegn Taye^b

^a Faculty of Computing and Informatics, Jimma Institute of Technology, Jimma, Ethiopia

^b Faculty of Civil and Environmental Engineering, Jimma Institute of Technology, Jimma, Ethiopia

*Corresponding author. E-mail: demeke.endalie@ju.edu.et

 DE, 0000-0002-2476-6217

ABSTRACT

Rainfall prediction is a critical task because many people rely on it, particularly in the agricultural sector. Rainfall forecasting is difficult due to the ever-changing nature of weather conditions. In this study, we carry out a rainfall predictive model for Jimma, a region located in south-western Oromia, Ethiopia. We propose a Long Short-Term Memory (LSTM)-based prediction model capable of forecasting Jimma's daily rainfall. Experiments were conducted to evaluate the proposed models using various metrics such as Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), Nash–Sutcliffe model efficiency (NSE), and R^2 , and the results were 0.01, 0.4786, 0.81 and 0.9972, respectively. We also compared the proposed model with existing machine-learning regressions like Multilayer Perceptron (MLP), k -Nearest Neighbors (KNN), Support Vector Machine (SVM), and Decision Tree (DT). The RMSE of MLP was the lowest of the four existing learning models i.e., 0.03. The proposed LSTM model outperforms the existing models, with an RMSE of 0.01. The experimental results show that the proposed model has a lower RMSE and a higher R^2 .

Key words: decision tree, KNN, LSTM, MLP, rainfall prediction, SVM

HIGHLIGHTS

- We propose a rainfall prediction model based on LSTM.
- An extensive experiment is used to present a detailed analysis of the proposed model.
- Contrasts with various predictive machine-learning algorithms.

INTRODUCTION

Almost 85% of Ethiopians live in rural areas and make their living through agriculture. Ethiopia's agricultural system is heavily dependent on rainfall. The forecast rainfall has a wide-ranging impact on agriculture as well as on travelers planning their trips. Predicting rainfall, on the other hand, is extremely difficult. A variety of factors influence rainfall, including humidity, maximum and minimum temperatures, wind speed and direction, and so on (Elwell & Stocking 1974; Danladi *et al.* 2018). The pattern of these parameters can be used to forecast rainfall. Machine-learning algorithms used for rainfall prediction include decision trees, k -nearest neighbors, linear regression, and rule-based methods (Ridwan *et al.* 2021). Deep learning can produce meaningful results for larger datasets. The primary goal of this research is to forecast rainfall using six basic rainfall parameters of maximum temperature, minimum temperature, relative humidity, solar radiation, wind speed and precipitation. Deep learning is used to create the predictive model. We propose an LSTM model for daily rainfall prediction.

In this study, we develop a predictive model (Xue *et al.* 2020) for Jimma in southwestern Oromia, Ethiopia. Jimma is the birthplace of the coffee Arabica (Mengistu *et al.* 2020). In short, too little water is never a good thing, and too much water can be either harmful or beneficial, depending on other environmental factors, to the coffee product. Despite numerous works on rainfall prediction using Artificial Neural Networks (ANN), MLP, and linear regression, there is no literature on deep-learning-based prediction applied to the same area in Jimma town (Liu *et al.* 2019). Since weather parameters vary from location to location, a model developed for one area would not be applicable to another.

The location was chosen because it is a coffee source, which helps the country earn money by exporting it. As a result of ineffective water resource management, the region is troubled by flooding and constant water scarcity (Carr 2001).

This is an Open Access article distributed under the terms of the Creative Commons Attribution Licence (CC BY 4.0), which permits copying, adaptation and redistribution, provided the original work is properly cited (<http://creativecommons.org/licenses/by/4.0/>).

Individuals who are aware of the upcoming day's rainfall can identify and mitigate water shortage problems and the occurrence of flooding. This study develops the ability to predict rainfall using a deep-learning model based on weather parameters recorded by the country's weather stations. The dataset for this study was gathered from the National Meteorological Service Agency (NMSA) of Ethiopia from the years 1985–2017 GC.

Rainfall estimation can be used for a variety of purposes, including reducing traffic accidents and congestion, increasing water management, reducing flooding, and so on. Meteorologists have long strived for weather forecasting that is both reliable and timely. Traditional theory-driven numerical weather prediction (NWP) approaches, on the other hand, face a slew of issues, including a lack of understanding of physical processes, difficulty extracting useful knowledge from a flood of observational data, and the need for powerful computational resources (Pu & Kalnay 2018). The successful implementation of data-driven deep-learning methods in a variety of fields, including computer vision, speech recognition, and time series prediction, has shown that deep-learning methods can effectively mine temporal and spatial features from temporal data. Meteorological data is an example of large geospatial data. Deep-learning-based weather prediction (DLWP) is expected to be a great asset to the conventional method (Hewage *et al.* 2021).

Rainfall forecasting is based on personal experience and observation of rainfall parameters. Machine-learning algorithms such as MLP have been used by researchers to predict rainfall. The ability of deep learning to predict rainfall is limited, particularly when using sensor-based datasets. MLP is the most popular neural network model for forecasting rainfall, according to recent surveys (Nayak *et al.* 2013; Sundaravalli & Geetha 2016; Ren *et al.* 2020). At present, many researchers have tried to introduce data-driven deep learning into weather forecasting, and have achieved some preliminary results. The following are some of the relevant works for this research.

Various researchers have proposed different research projects that use various machine-learning algorithms. In Lee *et al.* (2018) ANN was used to create a late spring–early summer rainfall forecasting model for the Geum River Basin in South Korea. The best ANN model with five input variables had relative root mean square errors of 25.84%, 32.72%, and 34.75% for training, validation, and testing datasets, respectively. The hit score, which is the number of hit years divided by the total number of years, was more than 60%, which indicates that the ANN model successfully predicts rainfall in the study area.

In Biyadglgn & Melkamu (2016) the authors proposed a rainfall predictive model for crop recommendation that can be used in some parts of Ethiopia. Their rainfall prediction model was created using ANN and KNN. The three basic rainfall parameters used were maximum temperature, minimum temperature, and average rainfall. They conducted experiments on summer rainfall using meteorological stations in Gojjam and Gonder. However, they did not forecast for all Ethiopian seasons, and their performance needs to be improved.

The authors at Dash *et al.* (2018) proposed a rainfall forecast for the Indian state of Kerala using KNN, ANN, and extreme learning algorithms. The rainfall prediction design model for Kerala presented here is crucial in addressing water shortages and preventing drought. They used time series meteorological data from the Indian Institute of Technology Madras (IITM). In terms of precision, the results show that ANN and Extreme Learning Machine (ELM) models outperform KNN models.

In Choubin *et al.* (2017) the authors present an ensemble forecast of semi-arid rainfall using large-scale climate predictors. They focused on computing the correlation between climate predictors and seasonal precipitation over a long-term forecast period (1967–2009) for a semi-arid catchment in Iran. Linear regression together with two nonlinear models, the adaptive neuro-fuzzy inference system (ANFIS) and the multi-layer perceptron, were applied to forecast seasonal ensemble precipitation time series. An ensemble forecast of spring precipitation modes showed a stronger correlation with the preceding season (winter predictors) in the ANFIS algorithm. An analysis suggests that seasonal precipitation is statistically aligned with the predictor's variability.

Climate modeling and prediction are critical in water resource management, particularly in arid and semi-arid countries where water shortages are common. The authors in Choubin *et al.* (2016) present a drought index modeling approach based on large-scale climate indices by using the Adaptive Neuro-Fuzzy Inference System (ANFIS), the M5P model tree, and the Multilayer Perceptron (MLP). They used factor analysis to determine the climate signal from 25 climate signals, and then used ANFIS, the M5P model tree, and MLP to forecast the Standardized Precipitation Index (SPI) one to 12 months in advance. The performance of the models was assessed using error parameters and Taylor diagrams, which revealed that the MLP outperformed the other models.

Interest in semi-arid climate forecasting has grown due to risks associated with above-average levels of precipitation. Longer-lead forecasts are difficult to make due to short-term extremes and data scarcity. The Classification and Regression Trees (CART) model, which is a rule-based algorithm, was used for prediction of the precipitation over a highly complex semi-arid climate system using climate signals. The work of Choubin *et al.* (2018) compared the accuracy of the CART model with

the two most commonly applied models, including time-series Auto-Regressive Integrated Moving Average (ARIMA) and ANFIS, for the prediction of precipitation. Various combinations of large-scale climate signals were considered as inputs. Their experimental results indicate that the CART model had better results (with Nash–Sutcliffe efficiency $NSE > 0.75$) compared with the ANFIS and ARIMA in forecasting precipitation.

The work of [Mishra *et al.* \(2018\)](#) suggested another rainfall prediction method based on an ANN model and a time series dataset. The implemented framework used documented time-series data from the Indian Meteorology Department in Pune to construct two models using a feed-forward neural network with a back-propagation algorithm (one-month-ahead prediction and two-months-ahead prediction). In the analysis of 3-25-1 regression, model 1 obtained the best results of 0.946 and 0.948 with validating and testing datasets respectively. Model 2 produced 0.913 and 0.910 for the validating and testing datasets, respectively, using 3-50-1 regression.

Deep-learning methods have advanced, and research has been done to apply them to time series prediction. The authors of [Sutskever *et al.* \(2014\)](#) proposed a multi-stacked LSTM to forecast temperature, wind speed, and humidity for 24 and 72 hours. They used hourly meteorological data from nine Morocco cities for 15 years, from 2000 to 2015. The authors concluded that deep LSTM networks could effectively forecast weather parameters and recommended that they can be used for other weather-related problems. In [Qiu *et al.* \(2017\)](#), the authors used multi-task CNN to forecast short-term precipitation in China using weather parameters obtained from several rain gauges. The authors came to the conclusion that multi-site features outperformed single-site features.

The authors at [Chhetri *et al.* \(2020\)](#) predict monthly rainfall over Simtokha, an area in Bhutan's capital, Thimphu. Bhutan's National Center of Hydrology and Meteorology Department (NCHM) provided the rainfall dataset. Based on the parameters reported by the automatic weather station in the area, they investigated the predictive capability of Linear Regression, Multi-Layer Perceptron (MLP), Convolutional Neural Network (CNN), LSTM, Gated Recurrent Unit (GRU), and Bidirectional Long Short-Term Memory (BLSTM). They proposed a model based on the BLSTM-GRU which outperforms the existing machine- and deep-learning models.

In reviewing the papers above, we discovered that using a deep-learning model to predict rainfall can improve its accuracy. As a result, we propose LSTM-based rainfall prediction to improve the accuracy of rainfall forecasting for Jimma, Ethiopia. This research focuses on the use of deep-learning techniques to forecast Jimma town rainfall using various parameters. The study's contribution can be summarized as follows:

1. We propose a rainfall prediction model based on LSTM.
2. An extensive experiment is used to present a detailed analysis of the proposed model.
3. Contrasts with various predictive machine-learning algorithms.

The remainder of the paper is structured as follows. The second section discusses the existing rainfall prediction models, and the third section discusses the methodology. The fourth section discusses the experimental results and their implications. Finally, in the fifth section, we conclude the work and make suggestions for future work.

MATERIALS AND METHODS

The proposed rainfall prediction model

The processing of the proposed rainfall prediction model begins with the collection of meteorological data. In this context, we use NMSA weather data. The collected data is then preprocessed, which includes things like eliminating empty entries, resolving missing values, and normalizing. The preprocessed data is provided to the deep-learning module to learn from it and predict rainfall for unseen data. [Figure 1](#) depicts the modules utilized in the proposed rainfall prediction model.

A description of each component of the proposed rainfall prediction architecture is given below.

Deep learning

Deep-learning techniques were used to forecast the rainfall in this paper. For the chosen location, Jimma, we proposed a deep-learning-based rainfall prediction model. The layers of the model and their functions are described below:

Input Layer: The input layer of a deep learning system is made up of artificial input neurons, which deliver the initial pre-processed weather data into the system for processing by subsequent layers of the neural neurons ([Mhatre *et al.* 2015](#)).

LSTM: LSTM is a special kind of Recurrent Neural Network (RNN), capable of learning long-term dependency ([Miao *et al.* 2020](#)). LSTMs were created expressly to address the issue of long-term dependency. They do not have to try very hard to

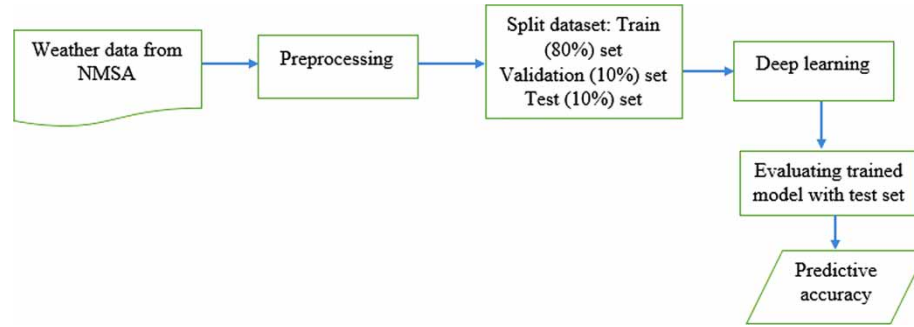


Figure 1 | Architecture of the proposed model of rainfall prediction.

retain information for long periods of time; it comes naturally to them. Although LSTMs have a chain-like structure, the repeating module differs. There are four neural network layers instead of one, each of which interacts differently (Kumar *et al.* 2018). Figure 2 depicts the modules and interaction of components of LSTM.

LSTM was used to take five basic weather parameters and predict the rainfall based on the input parameter value. The architecture was fixed after thoroughly hyper-tuning parameters of the LSTM. The hyperparameters were adjusted through heuristic knowledge of the programmer and randomized grid search.

Batch Normalization: Batch normalization is a deep neural network training strategy that standardizes each mini-input batch to a layer. This stabilizes the learning process and reduces the number of training epochs required to build deep networks dramatically (Schilling 2016).

Dense Layer: A dense network is formed when each neuron in a layer receives input from all of the neurons in the previous layer (Gelenbe & Yin 2017). It provides learning features from all the combinations of features from the previous layers.

Dropout: Dropout is a regularization method that approximates training a large number of neural networks with different architectures in parallel. During training, a number of layer outputs randomly drop out. This has the effect of making the layer look like and be treated like a different layer (Srivastava *et al.* 2014).

Activation Function: An activation function is a function that is added into an artificial neural network in order to help the network learn complex patterns. We used ‘*relu*’ as the activation function in our deep learning model, and it has been shown that this can be a powerful way to train the model.

Output Layer: The final layer of neurons in an artificial neural network is the output layer, which generates the program’s outputs. It is one in this case since it only yields one value.

Machine-learning method used

Multilayer perceptron

The best known and most widely used neural network is the multilayer MLP (Taravat *et al.* 2015). In this network model, the signal travels in only one direction, from input to output. The MLP neural network can be constructed using simple components. It can start with a single input neuron and grow to include multiple inputs. The stack of these neurons then

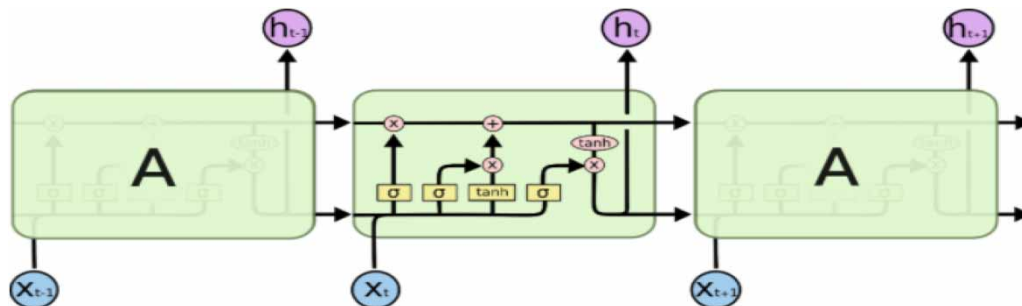


Figure 2 | The repeating module in an LSTM contains four interacting layers.

forms layers (Endalie & Tegegne 2021). In addition to processing units, a neural network is made up of direct weighted connections between neurons.

Neurons have an activation function that transforms the state of the previous layer's output to the next activation state based on the thresholding value. Each hidden layer processing unit takes the output of the previous layer's neurons as input and applies an activation function to it. The layer sends a numeric value to the next layer based on the threshold. MLP does not provide an increase in computing power over single-layer networks if the activation function is linear (Azadi *et al.* 2016). MLP's power is determined by the non-linear activation function. In this study, we used one hidden layer with 100 neurons. The other parameters we set were random state = 1, maximum iteration = 1,000, and the default value for the rest of the MLP regressor hyperparameters.

Decision tree

A Decision Tree (DT) is a machine-learning method for constructing a prediction model from data by partitioning the dataset and fitting a simple model to each partition (Song & Lu 2015). The goal of this algorithm is to create a model that predicts the value of a target variable, for which the decision tree uses the tree representation to solve the problem, where the leaf node corresponds to a class label and attributes are represented on the tree's internal node.

k -Nearest neighbor

The k -Nearest Neighbor (KNN) algorithm is one of the most widely used learning algorithms in machine-learning research (Garg & Pandey 2019). The basic idea behind KNN is to predict the label of a query instance based on the labels of the k closest instances in the stored data, assuming that the label of an instance is similar to that of its KNN instances. KNN is simple and easy to implement, but it is extremely effective in terms of prediction performance. In practice, the main difficulty with KNN is its high sensitivity to hyperparameter settings such as the number of nearest neighbors k , the distance function, and the weighting function.

The neighbors are taken from the dataset for which the classes (for k -NN classification) or the object property estimation (for k -NN regression) is known. This can be thought of as the training dataset for the calculation. The values for hyperparameters used in this study are neighbors = 3 and default values for the other parameters.

Support vector machine

Support Vector Machine (SVM) is a supervised machine-learning algorithm that can be used for classification or regression tasks (Bahari *et al.* 2014). It is, however, mostly used in classification problems. In the SVM algorithm, each data item is

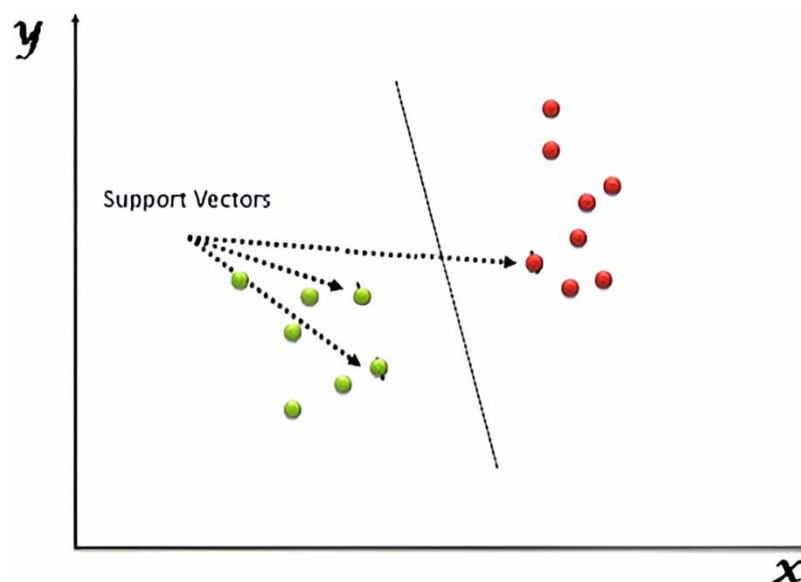


Figure 3 | Hyperplane that classifies points on the plane.

plotted as a point in n -dimensional space (where n is the number of features), with the value of each feature being the value of a specific coordinate. Then, classification is performed by locating the hyperplane that best distinguishes the classes as shown in Figure 3 below.

Dataset description

Jimma is a small town in Ethiopia’s southwestern Oromia region. The sensor data used in this study was obtained from a meteorology station in Jimma. It is located at 7°40’N 36°50’E latitude and longitude. The map in Figure 4 below depicts the study area.

Taking into account the length of the record, continuity of data and concurrent period of observation, the dataset used in this study is a daily record of weather parameters from 1985 to 2017. Prior to use, meteorological data were checked for consistency. There are six parameters in the dataset with 12,052 days of records of them. These parameters had zero or a small number of missing values, which were addressed during preprocessing. Weather parameters were extracted from weather data using the mean of maximum temperature, minimum temperature, relative humidity, solar radiation, wind speed, and precipitation. The first five parameters were used as inputs, and precipitation was used as an output. The dataset covers 33 years of records. In this study, we used a train-validate-test ratio of 80%, 10%, and 10%, respectively. We trained the model with data from 1985 to 2012, and then validated it with data from 2013 to the first half of 2015. The remaining data from the second half of 2015 to 2017 was used to evaluate the trained model’s performance. Table 1 below lists the weather parameters used in this study, as well as their measurement units. The pattern of the first five weather parameters is used to forecast precipitation.

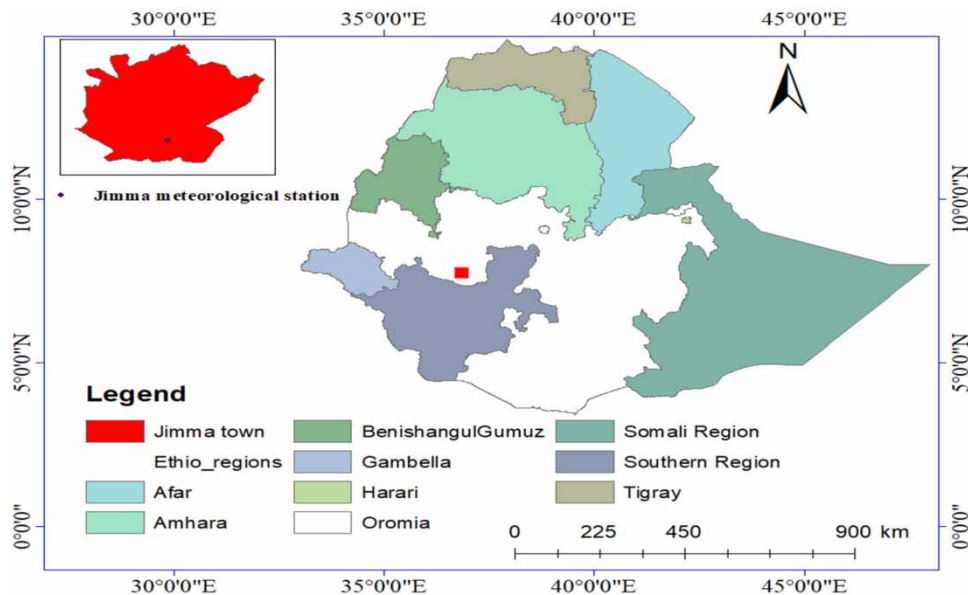


Figure 4 | The location of the study area.

Table 1 | Rainfall parameters used in our study and their corresponding measurement units

Rainfall parameters	Unit
Minimum temperature (t_{min})	°C
Maximum temperature (t_{max})	°C
Solar radiation	MJ m ⁻² day ⁻¹
Wind speed	Metres per second (m/s)
Relative humidity	Percentage (%)
Precipitation	Millimetres (mm)

Concordance correlation coefficient

The concordance correlation coefficient measures the agreement between two variables (Steichen & Cox 2002). Lin's concordance correlation coefficient (ρ_c) is a measure of how well a set of bivariate data (Y) compares to a 'gold standard' measurement or test (X). We can also compare two sets of measurements without a gold standard comparison. The procedure can be performed on datasets with 10 or more pairs. Lin's concordance correlation coefficient measures both precision (ρ) and accuracy (CB).

The value of ρ_c ranges from 0 to ± 1 . According to the authors of Nielsen *et al.* (2018), ρ_c values less than 0.9 are poor, 0.90–0.95 are moderate, 0.95–0.99 are significant, and 0.99 is almost excellent. Table 2 shows that the dependent variable (precipitation) has a poor concordance correlation with the five independent variables.

Data preprocessing

Data from NMSA are preprocessed in four states with arrow heads showing the flow of data. The dataset was cleaned up by removing the empty records in the weather parameters used for this study. The dataset's null values are standardized during the preprocessing phase. Missing data for precipitation were estimated using the linear regression method of XL STAT 2018 by considering correlation coefficients between variables. Missing data for daily minimum and maximum temperature, relative humidity, wind speed and sunshine hours were filled using the multiple imputation algorithm based on the Markov Chain Monte Carlo (MCMC) approach, also called fully conditional specification (van Buuren 2007). Initial values of the missing values were obtained sampled from a normal distribution with mean and standard error equal to the mean and standard error obtained on the available data, and for each variable of the dataset with missing values, an imputation method based on sampling and Ordinary Least Squares (OLS) regression was applied. The used model was a regression model with the studied variable as dependent variable and all the other variables as independent variables. Disturbance using data sampled from different distributions was also used. New imputed values were obtained using this model. We used a min–max scaler to normalize the weather parameters in order to obtain the new scaled value z . We used the min–max scaler on the data because the ranges of features differed (Li & Liu 2011):

$$z = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (1)$$

where x denotes the scaled value, $\max(x)$ denotes the maximum value, and $\min(x)$ denotes the minimum value from input x . The preprocessing step is depicted in Figure 5.

Table 2 | The concordance correlation coefficient between independent variables and dependent variable

Dependent variable	Independent variables				
	Concordance Correlation Coefficient (CCC)				
	Max_temp	Max_temp	Humidity	Solar	Wind speed
Precipitation	–0.015	0.147	0.0094	–0.041	0.0024

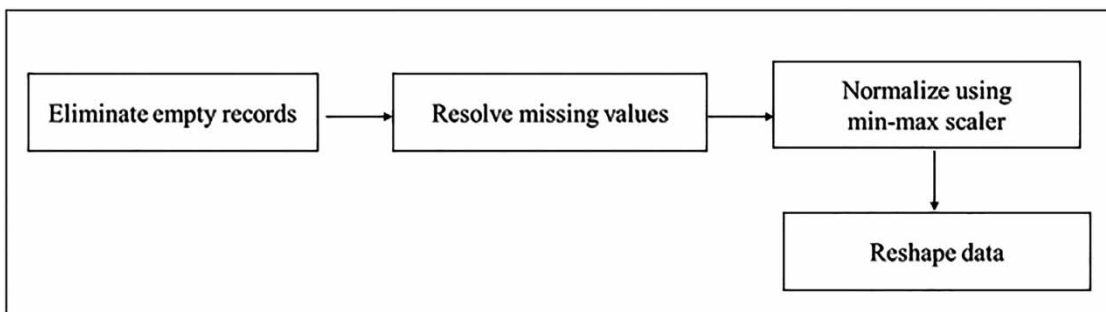


Figure 5 | Data preprocessing.

Reshaping data

The process of changing the dimension of the original data is known as data reshaping (Mishra *et al.* 2018). It is difficult to understand how to prepare the sequence data for input to an LSTM model. The definition of the input layer for the LSTM mode is commonly misunderstood. We transform the data sequence from a 2D matrix to the required 3D format of the LSTM input layer after eliminating empty records, resolving missing values, and using the min–max scaler.

Evaluation metrics

The study calculated the performance of the prediction model using RMSE, Normalized Root Mean Squared Error (NRMSE), NSE, Mean Absolute Error (MAE), MAPE and R^2 metrics. The formulas for RMSE, MAE, and R^2 are shown in Table 3 below.

RESULTS AND DISCUSSION

In this section, we investigate the performance of the proposed rainfall prediction model. All experiments were carried out in a Windows 10 environment on a machine equipped with a core i7 processor and 16 GB of RAM. The performance of the proposed model is compared with that of well-known machine-learning-based predictors such as MLP, SVM, KNN, and DT.

Results summary

The proposed model is evaluated with the six basic scoring metrics, i.e., RMSE, NRMSE, MAPE, MAE, NSE and R^2 . The results of the experiments are presented based on a 128-neuron LSTM model. In addition to using RMSE, NRMSE, MAPE, MAE, NSE and R^2 to assess the proposed deep-learning-based daily rainfall prediction model, we also assess the model's prediction accuracy using data that was not used during the training phase. Figure 6 below depicts the training and validation performance of the proposed model in terms of MAPE.

On the other hand, we evaluate the predictive performance of the proposed model using a testing dataset, and the results are presented in Table 4 below. The outcome demonstrates that the proposed model performs well because it mitigates all types of errors.

Figure 7 shows the results of the proposed (LSTM)-based rainfall prediction model for estimating rainfall for the following 60 days. The results show that the proposed methodology is 99.72% accurate in forecasting average rainfall (in mm). The red line on the graph represents the actual amount of average rainfall measured by the rain gauge, while the blue line represents the amount of average rainfall predicted by the proposed model. As a result, the proposed model can be used to predict rainfall of a specific day. The plot shows the actual daily rainfall values over Jimma collected from NMSA and predicted rainfall values for 60 days, where the x -axis and y -axis represent day and daily rainfall values respectively.

Table 3 | Evaluation metrics for daily rainfall prediction

Metrics	Formula
RMSE	$\left(\frac{1}{n} \sum_{i=1}^n (x_i - y_i)^2\right)^{1/2}$
NRMSE	$\frac{\text{RMSE}/\text{mean}}{\sum_{i=1}^n (y_i - x_i)^2}$
NSE	$1 - \frac{\sum_{i=1}^n (y_i - x_i)^2}{\sum_{i=1}^n (y_i - \text{mean of } y)^2}$
MAE	$\frac{1}{n} \sum_{i=1}^n x_i - x $
MAPE	$\frac{1}{n} \sum_{i=1}^n \frac{ A_i - F_i }{A_i}$
R^2	$1 - \frac{\text{Unexpected variation}}{\text{Total variation}}$

Where x_i is the model's simulated daily rainfall, y_i is the observed daily rainfall, A_i is the actual daily rainfall value, F_i is the forecast daily rainfall value, and n is the number of data points.

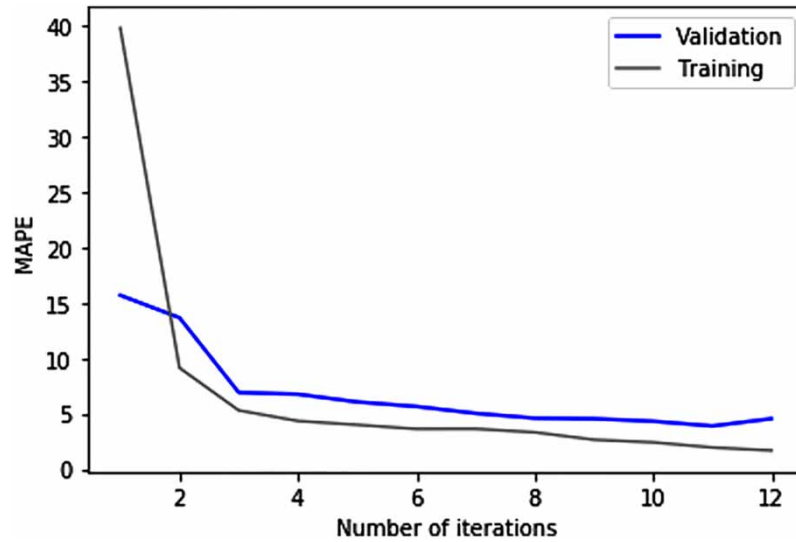


Figure 6 | Validation and training MAPE for the LSTM model as a function of the number of iterations.

Table 4 | Evaluation of the performance of the proposed model

Predictive model	Evaluation metrics					
	NRMSE	MAPE	RMSE	MAE	R^2 (%)	NSE
LSTM	0.018	0.4786	0.010	0.0082	99.72	0.81

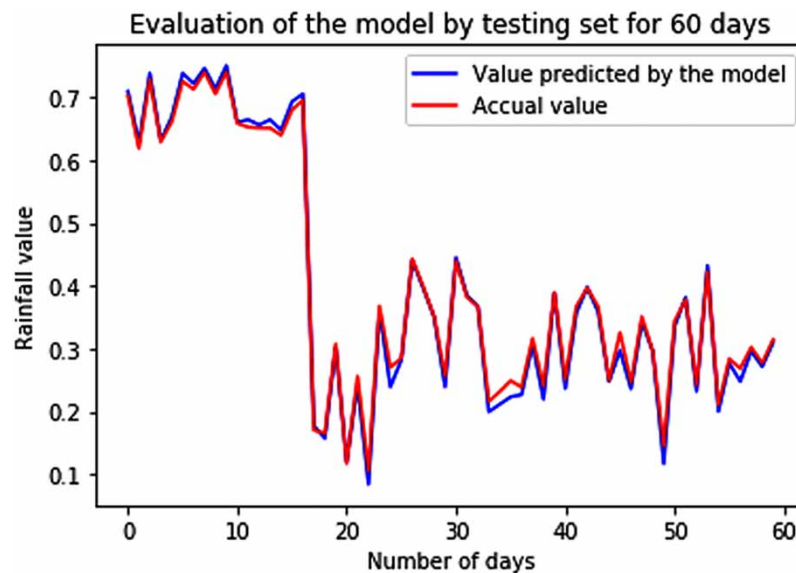


Figure 7 | Evaluation of the proposed model with testing dataset.

Comparative analysis

We compared our model with MLP, SVM, KNN (Taravat *et al.* 2015; Ayisha Siddiqua & Senthil Kumar 2019; Garg & Pandey 2019) and other methods on the NMSA dataset, as shown in Figure 8 in terms of RMSE. The proposed model

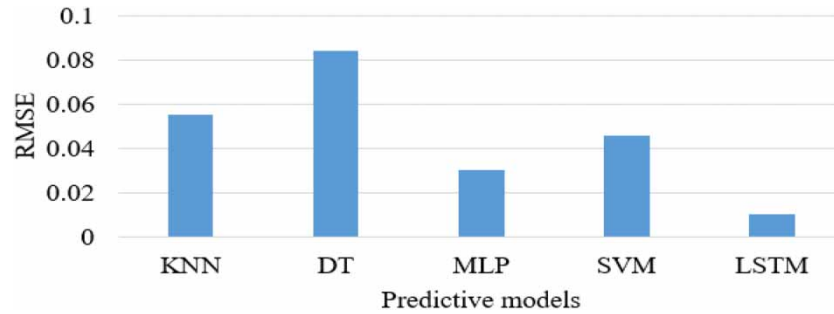


Figure 8 | RMSE values of four existing machine-learning models and the proposed model.

performed uniformly better than machine-learning techniques under study i.e., it reduced the RMSE from that of KNN, DT, MLP, and SVM by 4.5%, 7.4%, 2%, and 3.6%, respectively.

In addition, we also ran an experiment to show statistically significant differences between the results of the MLP, SVM, DT, KNN and LSTM methods when performing a k -fold cross-validation on the predictive models. The size k of the prediction algorithm above is 5. The results of this experiment are expressed in terms of RMSE, as shown in Table 5 below.

Table 5 indicates LSTM produced lower RMSE out of the total sample than MLP, SVM, DT, and KNN predictive algorithms. We compared LSTM with the four machine-learning algorithms by using the Wilcoxon Signed Rank Test. Results of the Wilcoxon Signed Rank Test for DT, MLP, KNN and SVM with LSTM are shown in Tables 6–9. The hypothesis used are H_0 : If the number of the positive difference is greater than the number of the negative difference, then the machine-learning algorithm is working better than LSTM; H_1 : If the number of the negative difference is greater than the number of the positive difference, then LSTM is working better than the machine-learning algorithm.

The proposed model clearly outperforms the current models in terms of RMSE when given the same tasks as shown in Figure 8. The total amount of rainfall (in mm) for the next day ($t + 1$) is the output of the designed model. Each day's weather features are included in each time-step. The time-step $T(n)$, for example, contains weather parameters for the n^{th} day. The

Table 5 | Comparison of KNN, SVM, MLP, DT and LSTM with k -fold cross-validation ($k = 5$)

k -Fold values	KNN	SVM	MLP	DT	LSTM
1	0.01185	0.02649	0.003159	0.0006308	0.0004753
2	0.0119	0.02603	0.002666	0.0004269	0.0006178
3	0.01292	0.0244	0.003165	0.003091	0.001304
4	0.0123	0.02609	0.001974	0.0003350	0.0001602
5	0.0120	0.02771	0.003307	0.000644	0.001229
Total out of sample	0.0122	0.02617	0.003173	0.00146	0.0008769

Table 6 | Wilcoxon Signed Rank Test difference between DT and LSTM

k -Fold	DT	LSTM	Difference	Positive	Difference	Rank	Signed rank
1	0.0006308	0.0004753	-0.0001555	-1	0.0001555	1	-1
2	0.0004269	0.0006178	0.0001909	1	0.0001909	3	3
3	0.003091	0.001304	-0.001787	-1	0.001787	5	-5
4	0.0003350	0.0001602	-0.0001748	-1	0.0001748	2	-2
5	0.000644	0.001229	0.000585	1	0.000585	4	4

Positive sum: 7; negative sum: 8; test statistics: 7.

The number of negative differences is greater than the positive differences, which indicates LSTM is working better than DT.

Table 7 | Wilcoxon Signed Rank Test difference between MLP and LSTM

k-Fold	MLP	LSTM	Difference	Positive	Difference	Rank	Signed rank
1	0.003159	0.0004753	-0.0026837	-1	0.0026837	5	-5
2	0.002666	0.0006178	-0.0020482	-1	0.0020482	3	-3
3	0.003165	0.001304	-0.001861	-1	0.001861	2	-2
4	0.001974	0.0001602	-0.0018138	-1	0.0018138	1	-1
5	0.003307	0.001229	-0.002078	-1	0.002078	4	-4

Positive sum: 0; negative sum: 15; test statistics: 0.

The result indicates, for all values of k -fold values, LSTM is working better than MLP.

Table 8 | Wilcoxon Signed Rank Test difference between KNN with LSTM

k-Fold	KNN	LSTM	Difference	Positive	Difference	Rank	Signed rank
1	0.01185	0.0004753	-0.0113747	-1	0.0113747	3	-3
2	0.0119	0.0006178	-0.0112822	-1	0.0112822	2	-2
3	0.01292	0.001304	-0.011616	-1	0.011616	4	-4
4	0.0123	0.0001602	-0.0121398	-1	0.0121398	5	-5
5	0.0120	0.001229	-0.010771	-1	0.010771	1	-1

Positive sum: 0; negative sum: 15; test statistics: 0.

The result indicates that for all values of k -fold values, LSTM works better than KNN. KNN does not produce better RMSE for the 5 folds than LSTM.

Table 9 | Wilcoxon Signed Rank Test difference between SVM with LSTM

k-Fold	SVM	LSTM	Difference	Positive	Difference	Rank	Signed rank
1	0.02649	0.0004753	-0.0260147	-1	0.0260147	4	-4
2	0.02603	0.0006178	-0.0254122	-1	0.0254122	2	-2
3	0.0244	0.001304	-0.023096	-1	0.023096	1	-1
4	0.02609	0.0001602	-0.0259298	-1	0.0259298	3	-3
5	0.02771	0.001229	-0.02648	-1	0.02648	5	-5

Positive sum: 0; negative sum: 15; test statistics: 0.

The result indicates that for all values of k -fold values, LSTM works better than SVM. SVM does not produce better RMSE for the 5 folds than LSTM.

proposed method (LSTM) with 128 neurons outperforms in terms of RMSE, MAPE, and R^2 with values of 0.01, 0.4786, and 0.9972 respectively.

The results shown in this section are consistent with state-of-the-art techniques. In this section, we compare the proposed model's results with those of previously utilized rainfall forecast techniques (Ayisha Siddiqua & Senthil Kumar 2019; Garg & Pandey 2019). As a result, we discovered that utilizing deep learning to construct a rainfall forecasting model enhances prediction accuracy. In addition to that, the proposed rainfall prediction model results in a 3.6% improvement in terms of RMSE over the method used in Endalie & Tegegne (2021), which is SVM, and 7.4% over the method used in Ayisha Siddiqua & Senthil Kumar (2019). The experimental result demonstrates that performance exceeds previously stated results.

The summary of comparisons of the proposed rainfall prediction model for Jimma town (source of the coffee Arabica) with previously utilized models of rainfall prediction (Choubin *et al.* 2016; Choubin *et al.* 2017; Choubin *et al.* 2018; Chhetri *et al.* 2020) introduced in the introduction section is shown in Table 10. The comparison is done in terms of RMSE.

The current study yielded the following broad guidelines for developing a rainfall forecast model: (1) selection of assessment measures to verify the prediction model's real efficacy; (2) usage of deep learning since it exceeds current well-known machine-learning regressor models. The following is an explanation of why performance in LSTM-based rainfall

Table 10 | Summary of comparison of the proposed model with previous studies

Evaluation metric	Prediction techniques used by previous studies					The propose prediction model LSTM
	MLP	ANFIS	BLSTM	ARIMA	BLSTM-GRU	
RMSE	0.03	0.074	0.04	0.057	0.016	0.01

prediction should be improved: (1) the LSTM has the capacity to delete or add information to the cell state using precisely regulated structures known as gates, and (2) it offers more attention to a selection of features to enhance prediction performance. Finally, based on the above findings, we can infer that using deep learning for time-series data prediction yields the best results in terms of numerous assessment metrics.

CONCLUSION

The study of deep-learning methods for rainfall prediction is presented, and a rainfall prediction model based on LSTM is proposed for Jimma in western Oromia, Ethiopia. The dataset for the experiment was gathered from NMSA of Ethiopia. The dataset includes daily records of weather parameters such as t_{max} , t_{min} , relative humidity, solar radiation, wind speed, and precipitation from 1985 to 2017. On this dataset, several experiments and comparisons with the existing machine-learning-based model are performed to validate the performance of the proposed predictive model. The experimental results demonstrate that the proposed predictive model produces a promising outcome. As a result, the proposed LSTM-based rainfall predictive model is suitable for use in a variety of applications requiring rainfall prediction, such as smart agriculture. In the future, we aim to develop a rainfall prediction model that includes sea-surface temperature, global wind circulation, and climate indices, as well as to investigate the impact of climate change on rainfall.

ACKNOWLEDGEMENTS

This study was performed by three academic staff at Jimma Institute of Technology, Jimma University, Ethiopia. The authors would like to thank the institute for its assistance with various resources, as well as NMSA of Ethiopia for providing the dataset for our experiments. The authors would like to thank Jimma University for support during the research work.

CONFLICT OF INTEREST

The authors declare that they do not have any conflicts of interest with regard to this work.

FUNDING

This study received no outside funding.

DATA AVAILABILITY STATEMENT

All the relevant data are uploaded on GitHub and accessible via the following URL: <https://github.com/demekeendalie/rainfall-prediction>.

REFERENCES

- Ayisha Siddiqua, L. & Senthil Kumar, N. C. 2019 Heavy rainfall prediction using Gini index in decision tree. *International Journal of Recent Technology and Engineering (IJRTE)* 8 (4), 4558–4562.
- Azadi, S., Amiri, H. & Reza Rakhshandehroo, G. 2016 Evaluating the ability of artificial neural network and PCA-M5P models in predicting leachate COD load in landfills. *Waste Management* 55, 220–230.
- Bahari, N. I. S., Ahmad, A. & Aboobaidar, B. M. 2014 Application of support vector machine for classification of multispectral data. In: *7th IGRSM International Remote Sensing & GIS Conference and Exhibition*, 22–23 April, Kuala Lumpur, Malaysia.
- Biyadglgn, Y. & Melkamu, H. 2016 *Rainfall Prediction and Cropping Pattern Recommendation Using Artificial Neural Network: A Case Study for Ethiopia*. Addis Ababa University, Addis Ababa, Ethiopia.
- Carr, M. K. V. 2001 The water relations and irrigation requirements of coffee. *Experimental Agriculture* 37 (1), 1–36.
- Chhetri, M., Kumar, S., Roy, P. P. & Kim, B.-G. 2020 Deep BLSTM-GRU model for monthly rainfall prediction: a case study of Simtokha, Bhutan. *Remote Sensing* 12, 3174.

- Choubin, B., Malekian, A. & Golshan, M. 2016 Application of several data-driven techniques to predict a standardized precipitation index. *Atmosfera* **29** (2), 121–128.
- Choubin, B., Malekian, A., Samadi, S., Khalighi-Sigaroodi, S. & Sajedi-Hosseini, F. 2017 An ensemble forecast of semi-arid rainfall using large-scale climate predictors. *Meteorological Applications* **24** (3), 376–386.
- Choubin, B., Zehtabian, G., Azareh, A., Rafiei-Sardooi, E., Sajedi-Hosseini, F. & Kişi, Ö. 2018 Precipitation forecasting using classification and regression trees (CART) model: a comparative study of different approaches. *Environmental Earth Sciences* **77**, 314.
- Danladi, A., Stephen, M., Aliyu, B. M., Gaya, G. K., Silikwa, N. W. & Machael, Y. 2018 Assessing the influence of weather parameters on rainfall to forecast river discharge based on short-term. *Alexandria Engineering Journal* **57** (2), 1157–1162.
- Dash, Y., Mishra, S. K. & Panigrahi, B. K. 2018 Rainfall prediction for the Kerala state of India using artificial intelligence approaches. *Computers & Electrical Engineering* **70**, 66–73.
- Elwell, H. A. & Stocking, M. A. 1974 Rainfall parameters and a cover model to predict runoff and soil loss from grazing trials in the Rhodesian sandveld. *Proceedings of the Annual Congresses of the Grassland Society of Southern Africa* **9** (1), 157–164.
- Endalieu, D. & Tegegne, T. 2021 Designing a hybrid dimension reduction for improving the performance of Amharic news document classification. *PLoS ONE* **16** (5), e0251902.
- Garg, A. & Pandey, H. 2019 Rainfall prediction using machine learning. *International Journal of Innovative Science and Research Technology* **4** (5), 56–58.
- Gelenbe, E. & Yin, Y. 2017 Deep learning with dense random neural networks. In: *5th International Conference on Man–Machine Interactions, ICMMI, 17, 3–6 October, Cracow, Poland*.
- Hewage, P., Trovati, M. W., Pereira, E. & Behera, A. 2021 Deep learning-based effective fine-grained weather forecasting model. *Pattern Analysis and Applications* **24** (1), 343–366.
- Kumar, J., Goomer, R. & Singh, A. K. 2018 Long short term memory recurrent neural network (LSTM-RNN) based workload forecasting model for cloud datacenters. *Procedia Computer Science* **125**, 678–682.
- Lee, J., Kim, C.-G., Lee, J. E., Kim, N. W. & Kim, H. 2018 Application of artificial neural networks to rainfall forecasting in the Geum River Basin, Korea. *Water* **10**, 1448.
- Li, W. & Liu, Z. 2011 A method of SVM with normalization in intrusion detection. *Procedia Environmental Sciences* **11**, 256–262.
- Liu, Q., Zou, Y., Liu, X. & Linge, N. 2019 A survey on rainfall forecasting using artificial neural network. *International Journal of Embedded Systems* **11** (2), 240–248.
- Mengistu, M. W., Workie, M. A. & Mohammed, A. S. 2020 Physical and cup quality attributes of Arabica coffee (*Coffea arabica* L.) varieties grown in highlands of Amhara Region, northwestern Ethiopia. *International Journal of Agronomy* **2020**, 6420363.
- Mhatre, M. S., Siddiqui, F., Dongre, M. & Thakur, P. 2015 A review paper on artificial neural network: a prediction technique. *International Journal of Scientific & Engineering Research* **6** (12), 161–163.
- Miao, K.-c., Han, T.-t., Yao, Y.-q., Lu, H., Chen, P., Wang, B. & Zhang, J. 2020 Application of LSTM for short term fog forecasting based on meteorological elements. *Neurocomputing* **408**, 285–291.
- Mishra, N., Soni, H. K., Sharma, S. & Upadhyay, A. K. 2018 Development and analysis of artificial neural network models for rainfall prediction by using time-series data. *International Journal of Intelligent Systems and Applications* **10** (1), 16–23.
- Nayak, D. R., Mahapatra, A. & Mishra, P. 2013 A survey on rainfall prediction using artificial neural network. *International Journal of Computer Applications* **72** (16), 32–40.
- Nielsen, P. P., Fontana, I., Sloth, K. H., Guarino, M. & Blokhuis, H. 2018 Validation and comparison of 2 commercially available activity loggers. *Journal of Dairy Science* **101** (6), 5449–5453.
- Pu, Z. & Kalnay, E. 2018 Numerical weather prediction basics: models, numerical methods, and data assimilation. In: *Handbook of Hydrometeorological Ensemble Forecasting* (Q. Duan, F. Pappenberger, A. Wood, H. L. Cloke & J. C. Schaake, eds), Springer, Berlin, Germany.
- Qiu, M., Zhao, P., Zhang, K., Huang, J., Shi, X., Wang, X. & Chu, W. 2017 A short-term rainfall prediction model using multi-task convolutional neural networks. In: *2017 IEEE International Conference on Data Mining (ICDM)*, IEEE, Piscataway, NJ, USA, pp. 395–404.
- Ren, X., Li, X., Ren, K., Song, J., Xu, Z., Deng, K. & Wang, X. 2020 Deep learning-based weather prediction: a survey. *Big Data Research* **23**, 100178.
- Ridwan, W. M., Sapitang, M., Aziz, A., Kushiar, K. F., Ahmed, A. N. & El-Shafie, A. 2021 Rainfall forecasting model using machine learning methods: case study Terengganu, Malaysia. *Ain Shams Engineering Journal* **12** (2), 1651–1663.
- Schilling, F. 2016 *The Effect of Batch Normalization on Deep Convolutional Neural Networks*. Master's thesis, KTH, Stockholm, Sweden.
- Song, Y.-y. & Lu, Y. 2015 Decision tree methods: applications for classification and prediction. *Shanghai Archives of Psychiatry* **27** (2), 130–135.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. 2014 Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* **15** (1), 1929–1958.
- Steichen, T. J. & Cox, N. J. 2002 A note on the concordance correlation coefficient. *The Stata Journal* **2** (2), 183–189.
- Sundaravalli, N. & Geetha, A. 2016 A study & survey on rainfall prediction and production of crops using data mining techniques. *International Research Journal of Engineering and Technology (IRJET)* **3** (12), 1269–1274.

- Sutskever, I., Vinyals, O. & Le, Q. V. 2014 Sequence to sequence learning with neural networks. In: *NIPS'14: Proceedings of the 27th International Conference on Neural Information Processing Systems*, Volume 2 (Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence & K. Q. Weinberger, eds), MIT Press, Cambridge, MA, USA, pp. 3104–3112.
- Taravat, A., Proud, S., Peronaci, S., Del Frate, F. & Oppelt, N. 2015 [Multilayer perceptron neural networks model for Meteosat Second Generation SEVIRI daytime cloud masking](#). *Remote Sensing* **7**, 1529–1539.
- van Buuren, S. 2007 [Multiple imputation of discrete and continuous data by fully conditional specification](#). *Statistical Methods in Medical Research* **16**, 219–242.
- Xue, Y., Jiang, J. & Hong, L. 2020 [A LSTM based prediction model for nonlinear dynamical systems with chaotic itinerancy](#). *International Journal of Dynamics and Control* **8**, 1117–1128.

First received 28 September 2021; accepted in revised form 6 November 2021. Available online 17 November 2021