

# JADAVPUR UNIVERSITY

Department of Computer Science and Engineering  
Kolkata, West Bengal, India 2023

---

## Deep Learning Based Long-Term Rainfall Forecasting for Meteorological Subdivisions In India

---

By

**Ritam De**

REGN. NO. 154196 - OF 2020-2021

EXAM ROLL NO. -M6TCT23029

under the supervision of

**Dr. Sarbani Roy**

Professor

*Thesis submitted in partial fulfilment of requirements  
for the degree of*

MASTER OF TECHNOLOGY IN COMPUTER SCIENCE &

TECHNOLOGY

OF JADAVPUR UNIVERSITY



# Faculty of Engineering and Technology Jadavpur University

## Certificate of Recommendation

---

This is to certify that the work embodied in this thesis entitled "**Deep Learning Based Long Term Rainfall Forecasting for Meteorological Subdivisions in India**" has been satisfactorily completed by **Ritam De** (Registration Number 154196 of 2020-21; Class Roll No. 002010504031; Examination Roll No. M6TCT23029). It is a bona-fide piece of work carried out under my supervision and guidance at Jadavpur University, Kolkata for partial fulfilment of the requirements for the awarding of the **Master of Technology in Computer Science and Technology** degree of the Department of Computer Science and Engineering, Faculty of Engineering and Technology, Jadavpur University, during the academic year 2020-23.

---

Dr. Sarbani Roy,  
Professor,  
Department of Computer Science and Engineering,  
Jadavpur University.  
(Supervisor)

Countersigned by

---

**Prof. Nandini Mukherjee**

Head, Department of Computer Science and Engineering,  
Jadavpur University, Kolkata-32

---

**Prof. Saswati Mazumdar**

Dean, FET  
Faculty of Engineering and Technology  
Jadavpur University



**Department of Computer Science and Engineering  
Faculty of Engineering and Technology  
Jadavpur University, Kolkata - 700 032**

## **Certificate of Approval**

This is to certify that the thesis entitled "**Deep Learning Based Long Term Rainfall Forecasting for Meteorological Subdivisions in India**" is a bona-fide record of work carried out by **Ritam De** (Registration Number 154196 of 2020-21; Class Roll No. **002010504031**; Examination Roll No. M6TCT23029) in partial fulfilment of the requirements for the award of the degree of **Master of Technology in Computer Science and Technology** in the Department of Computer Science and Engineering, Jadavpur University, during the period of Oct 2020 to June 2023. It is understood that by this approval, the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the thesis only for the purpose of which it has been submitted.

Examiners:

(Signature of The Examiner)

(Signature of The Supervisor)



**Department of Computer Science and Engineering  
Faculty of Engineering and Technology  
Jadavpur University, Kolkata - 700 032**

**Declaration of Originality  
and Compliance of Academic Ethics**

I hereby declare that the thesis entitled "**Deep Learning Based Long Term Rainfall Forecasting for Meteorological Subdivisions in India**" contains literature survey and original research work by the undersigned candidate, as a part of his degree of **Master of Technology in Computer Science and Technology** in the Department of Computer Science and Engineering, Jadavpur University. All information has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

**Name:** Ritam De

**Examination Roll No.:** M6TCT23029

**Registration No.:** 154196 of 2020-21

**Thesis Title:** Deep Learning Based Long Term Rainfall Forecasting for Meteorological Subdivisions in India

**Signature of the Candidate:**

# ACKNOWLEDGEMENT

I am pleased to express my gratitude and regards towards my Project Guide **Dr. Sarbani Roy**, Assistant Professor, Department of Computer Science and Engineering, Jadavpur University, without whose valuable guidance, inspiration, and attention towards me, pursuing my project would have been impossible.

Last but not the least, I express my regards towards my friends and family for bearing with me and for being a source of constant motivation during the entire term of the work.

-----  
**Ritam De**

MTCT Final Year

Exam Roll No. - M6TCT23029

Regn. No. - 154196 of 2020-21

Department of Computer Science and Engineering,  
Jadavpur University.

# JADAVPUR UNIVERSITY

## *Abstract*

Faculty of Engineering and Technology  
Computer Science and Engineering

Master of Technology

**Deep Learning Based Long Term Rainfall Forecasting for Meteorological Subdivisions in  
India**

By **Ritam De**

Rainfall forecasting is very important because heavy and irregular rainfall can have many impacts like destruction of crops and farms, damage of property so a better forecasting model is essential for an early warning that can minimize risks to life and property and managing the agricultural farms in better way. This prediction mainly helps farmers and water resources can be utilized efficiently. Rainfall prediction is a challenging task and the results should be accurate. A good forecast of rainfall is essential for proper agricultural investment. Prediction of time series data in meteorology can assist in decision-making processes carried out by organizations responsible for the prevention of disasters. This paper presents Multi-layer Long Short-Term Memory (Multi-layer LSTM) based Recurrent Neural Network (RNN), Functional Transduction and Conformer model to predict rainfall. The neural network is trained and tested using a standard dataset of rainfall. The parameters considered for the evaluation of the performance and the efficiency of the proposed rainfall prediction model are Root Mean Square Error (RMSE), accuracy, number of epochs, loss, and learning rate of the network.

**Keywords:** Time series data, Time series pattern, Rainfall Pollution



# Contents

1. Introduction .....	17
1.1 Overview.....	17
1.2 Motivation .....	17
1.3 Contribution .....	18
1.4 Organization .....	19
2. Literature Review .....	21
3. Methodology .....	25
3.1 Problem Statement.....	26
3.2 A Workflow .....	26
3.2.1 Module Description .....	26
Data Collection .....	27
Data Pre-processing .....	27
Prepare Dataset to Fit into the Model .....	28
Monthly basis rainfall of each state based on deep learning model .....	29
3.3 Rainfall Data Collection .....	30
3.4 Proposed Approach .....	31
4. Results and Analysis .....	36
4.1 Experimental Setup .....	36
4.2 Results.....	36
4.2.1 Result Analysis for Rainfall Prediction .....	37
5. Conclusion and Future Work .....	44
5.1 Conclusion .....	44
5.2 Future Work .....	45



## List of Figures

- 3.1: Overall Architecture of the Work
- 3.2: Dataset after removing Unnecessary Information
- 3.3: Exponential Moving Average (EMA) shows trending rainfall data
- 3.4: Monthly precipitation rates for every state
- 3.5 Conformer model architecture
- 3.6 Functional transformer architecture
- 3.7 Computation Flow Diagram of Long Short-Term Memory Model
- 4.1 Experimental results for rainfall data of states using Functional transformer model
- 4.2 Experimental results for rainfall data of states using Conformer model
- 4.3 Experimental results for rainfall data of states using LSTM model

## List of Tables

3.1: Subdivision name, months, and rainfall percentages

3.2: Monthly precipitation rates for every state

4.2.1 Presents the experimental results of prediction using Functional Transformer

4.2.2 Presents the experimental results of prediction using Conformer

4.2.3 Presents the experimental results of prediction using LSTM





# Chapter 1

## Introduction

### 1.1 Overview

In meteorological departments, the accuracy of precipitation forecasts continues to be a key research topic. Predicting heavy rainfall and averting calamities are two benefits of accurate rainfall forecasting. It mostly has to do with the economy and human existence. In order for people and governments to be prepared for any disaster, effective forecasting techniques are required to detect droughts and floods earlier. Rainfall forecasting in India is very important because agriculture is the main sector of our economy. Artificial intelligence, which includes machine learning, is used to address numerous issues in information technology. A typical use of machine learning is to forecast a result based on historical data. Because of the numerous natural disasters that might result from severe rainfall, rainfall forecasting is crucial. By seeing a trend of different rainfall levels over time at different stations, we are attempting to identify the features of these amounts of rainfall in each state and relate their behaviour. We track the rainfall in several states using the results of this investigation.

We collect the data of different amount of rainfall of different states from the government of India website. The dataset that we get from the website is time series data of almost 100 years' Rainfall data and data of rainfall that are taken at every month. And from the time series data we try to find the repeated pattern of different granularity (monthly and yearly). There are several machine learning, deep learning and statistical methods which discover repeated patterns from time series data. The purposes of this research are to develop a highly accurate and adaptable statistical model for predicting univariate rainfall in each state of India as well as to investigate how intermediate rainfall affects the precision of forecasts made with the help of conformer model, multi-layer LSTM model, and functional transduction models.

### 1.2 Motivation

Rainfall is a major severe phenomenon within the climatic system. It has a straight influence on ecosystems, water, resources, management, and agriculture. Rainfall is the main source for the people whose entire life depends on water. There are few states in India like Kerala,

Karnataka, Goa, Orissa, West Bengal, Arunachal Pradesh, Sikkim, and Uttarakhand that have heavy rainfall—around 3000 mm to 1500 mm per year—which helps farmers to cultivate their crops without bothering about the deficiency of water resources. But few other states like Uttar Pradesh, Rajasthan and Gujarat receive very low rainfall—less than 600 mm per year—which is a major cause for drought in those areas. Sometimes, heavy rainfall also leads to floods in states like Orissa and Kerala, which affects people by severely damaging their properties. As a result, the economy of these states goes down for a certain period, it takes a lot of time for the people to recover their original lifestyle and to carry out their occupation. Nearly 1400 people were dead due to flood in various places of India in the year 2018, so that leads to loss of human life too.

On the other hand, the major economy of a country depends on food production, which must be enough for the total population of the country; if so, then the dependency on other countries for the importation of food products is reduced. India receives 70% of the rainfall for irrigation during monsoon season, and due to increase in temperature and evapotranspiration, the water is not enough to use for the whole year for agriculture. In this approach, analysing past rainfall data and anticipating future precipitation become crucial parts of forecasting and analysing rainfall data, which may aid in helping people find a solution when there is a deficit or surplus of precipitation.

### 1.3 Contribution

We use statistical data from India to build our model, which can discover the amount of rainfall in different areas of India. Predicting rainfall can be a boon in many situations, like crop irrigation, planning power production, and educating the people about the rainfall of that region. In the last decade, there has been some research on weather forecasting using neural networks (NN), which can be found in the literature. NN model for predicting rainfall a Recurrent Neural Network (RNN) model for predicting regional annual runoff. A fuzzy time series model for temperature prediction; an ensemble of NN models for predicting temperature, wind speed, and humidity; chaotic oscillatory-based NN for short-term wind forecasting using LIDAR data NN fuzzy wavelet model for long-term rainfall forecasting. We demonstrate the flexibility and efficiency of our model for fitting function-valued operators. In this study, the amount of rainfall that falls in each Indian state annually is used to model and predict precipitation using three neural network models: transformer, conformer, and multilayer LSTM. Also compared are their disparities in computation speed and accuracy. The open-source dataset is where the data came from. Determine the classification of precipitation into each month and precipitation with a range of 0 to 500 mm. Clouds have a significant impact on precipitation, even though it is well recognised that they do not entirely influence precipitation. In fact, it is possible to at least generate hazy predictions for some gridded areas using the dataset.

Of course, it is undeniable that, regardless of the model used, the complexity of the weather makes It is difficult to achieve a high level of prediction from rainfall information alone, but it is sufficient to complete a preliminary evaluation of both models and provide ideas for subsequent studies.

## 1.4 Organization

The thesis is organized as follows: Chapter 2 describe some existing work related to precipitation discovery using neural network models and some statistical method. In Chapter 3 we have discussed the problem statement mathematically and graphically present workflow of the thesis and discuss each part. In this chapter we have also discussed how rainfall data is collected and described the model that is used, parameters that can be tuned and the algorithms those helps to find repeated pattern. Chapter 4 discusses the experimental setup in which we built our model and results that we get. Finally, Chapter 5 concludes the thesis and future scope of our work.



## Chapter 2

# Literature Review

In this chapter, we go through the specifics of previous research on predicting rainfall using a machine learning model. Numerous indicators of a person's personal and professional life can be monitored. As shown, for instance, in seismic investigations and finance, sensor technology is always evolving, and sensor counts are rising. Knowledge discovery is challenging as a result of the enormous amounts of complicated data produced, which are typically in the form of time series. To imitate the complex mathematical expressions of physical processes of rainfall, for last two decades, machine learning methods are best in advancement of prediction systems which are provided with better performance and very reasonable solutions. due to more benefits and potential of ML, the popularity of ML is highly increased among hydrologists. Researchers introducing novel ML methods and hybridizing of the existing ones aim at discovering more effective and accurate prediction models. In a recent paper [1] DLWP, a Deep-Learning Weather Prediction Models algorithm is introduced, This model can recursively predicts six key atmospheric variables with six-hour time resolution. Convolutional neural networks (CNNs) on a cubed sphere grid are used in this computationally effective approach to provide global forecasts. On a single GPU, the trained model can generate a 320-member set of six-week forecasts at 1.4° resolution in under three minutes. A set of 32 DLWP models with slightly varying learned weights is created by randomising the CNN training process, which is the main method used to construct ensemble spread.

Several related works can be found in [2] [3] [4]. In [2], the authors present an integrated forecast of semi-arid precipitation using large-scale climate predictions. They focused on calculating the correlation between climate predictions and seasonal rainfall over a long-term forecast period (1967–2009) for a semi-arid watershed in Iran. Linear regression together with two nonlinear models, adaptive neuro-fuzzy inference system (ANFIS) and multi-layer perceptron, were applied to predict seasonal ensemble precipitation time series. An integrated forecast of the spring precipitation mode in the ANFIS algorithm showed a strong correlation with the previous season (winter predictors). An analysis suggests that seasonal precipitation is statistically correlated with predictor variability. Climate modelling and prediction are important in water resources management, especially in arid and semi-arid countries where water scarcity is common.

Deepali Patil et.al. (2020) [3]. state that weather and precipitation, in general, are highly nonlinear and complex. Events that require advanced computer modelling and simulation for accurate prediction. to guess various machine learning models are used including precipitation, multiple linear regression, neural Networks, K-means, Naive Bayes and others. Extraction, training, and testing of data sets and identification and Rainfall forecasting, these systems accomplish one of these applications. This paper shows multiple uses Linear Regression and Neural Networks for Rainfall Forecasting and Decision Tree Algorithms for Recommendations Crops Thus, it is concluded that we can reasonably predict rainfall and suggest crops.

Unsupervised semantic segmentation is a well-researched issue in the time series area. Current methodologies have several flaws, including the implicit assumption that all data can be segmented, which causes problems when that assumption is false, and the requirement that most approaches require many parameters to be set or learned. These limitations have limited the use of time-series semantic segmentation outside of academic contexts. According to Akash Parmar et.al (2017) [4] forecasting of heavy rainfall is a major concern for Meteorological Department. As it is directly related to economy and human existence. It causes natural calamities e.g., Floods and droughts that affect people around the world every year. Precipitation forecasting is critical to accuracy countries like India, whose economy is heavily dependent on agriculture. Statistical techniques fail to provide better the accuracy of rainfall forecasts is due to the dynamic character of the atmosphere. Due to its non-linearity for rainfall data, artificial neural network is a superior technique. In a tabular format, review tasks and comparison of different methods and algorithms employed by researchers for rainfall forecasting. The aim of this presented paper is to make the techniques and methods used in this research accessible to non-specialists.

Li et al. (2020) [5] proposed a Functional Transformer-based model for daily rainfall prediction. They compared the Functional Transformer model with traditional statistical methods and other deep learning models, including LSTM and GRU. The Functional Transformer model outperformed other models in terms of forecasting accuracy and captured the non-linear relationships and temporal patterns in rainfall data effectively. The functional operations embedded within the Functional Transformer model facilitated the modelling of complex dependencies and improved the forecasting performance.

In another study, Wu et al. (2021) [6] applied the Functional Transformer model to multi-step ahead rainfall prediction. They explored different variations of the Functional Transformer architecture and evaluated their performance on a large-scale rainfall dataset. The Functional Transformer models demonstrated superior performance compared to traditional methods and other deep learning models. The ability of the Functional Transformer to capture non-linear relationships and long-term dependencies contributed to its success in accurately predicting future rainfall values.

In another study, Zhang et al. (2022) [7] applied the Conformer model to monthly rainfall prediction. They used a large-scale dataset and compared the Conformer model with traditional methods and other deep learning models. The Conformer model outperformed other models in terms of forecasting accuracy, especially for capturing complex patterns associated with seasonal variations and climate phenomena. The attention mechanism in the

Conformer model allowed it to focus on relevant features and capture subtle dependencies in the rainfall time series.

Chen et al. (2023) [8] explored the Conformer model for long-term rainfall forecasting. They compared the Conformer model with other deep learning models, including LSTM and Transformer. The Conformer model demonstrated competitive performance, effectively capturing the temporal dependencies and complex patterns in long-term rainfall data. Its self-attention mechanism allowed for flexible modelling of interdependencies, leading to improved forecasting accuracy.

In a study by Liu et al. (2020) [5], LSTM models were used for seasonal rainfall forecasting. They explored the use of LSTM models to capture the seasonal patterns and dependencies in rainfall data. The LSTM models demonstrated promising results, effectively capturing the long-term dependencies and seasonal variations, thus enabling accurate seasonal rainfall forecasts

Previous research works show that, Researchers have used machine-learning algorithms such as MLP to predict rainfall. The ability of deep learning to predict rainfall is limited, especially when sensor-based datasets are used. MLP is the most popular neural network model for rainfall forecasting according to recent studies. Currently, many researchers have attempted to introduce data-driven deep learning into weather forecasting and have achieved some preliminary results. Accurate rainfall prediction has become very complicated in recent times due to climate change and variability. The efficiency of classification algorithms in rainfall prediction has flourished. The study contributes to using various classification algorithms for rainfall prediction in the different time zones of different region. The classification algorithms include Decision Tree (DT), Random Forest (RF), Multilayer Perceptron (MLP), Extreme Gradient Boosting (XGB) and K-Nearest Neighbour (KNN), Long short-term memory (LSTM). The performance of the classification algorithms was examined based on precision, recall, f1-score, accuracy, and execution time with various training and testing data ratios. On all three training and testing ratios: 70:30, 80:20 and 90:10, RF, XGB and MLP performed well, whereas KNN performed least across all zones. In terms of the execution time of the models, Decision Tree is consistently portrayed as the fastest, whereas MLP used the most run time.



## Chapter 3

# Methodology

This chapter is basically graphically present and describes the overall process of the thesis work. The problem statement of the work is presented mathematically and repeated pattern in different time series data with different states. This chapter also describes how rainfall data is collected and how to pre-process the dataset to fit into the model of finding the repeated pattern.

### 3.1 Problem Statement

Accurate and timely long-term rainfall forecasting plays a critical role in various sectors such as agriculture, water resource management, and disaster preparedness. Traditional rainfall prediction methods often rely on statistical models that have limited capacity to capture complex spatiotemporal patterns and dependencies present in rainfall data. Deep learning-based approaches have shown promising results in various domains, but their potential for long-term rainfall forecasting remains largely untapped. The dataset collected from the central government website consists of the rainfall records of different stations in various states of India. This dataset includes multiple states that collect rainfall data. It is organized into a set  $P = \{P_1, P_2, \dots, P_m\}$  representing the twenty-eight states and a set  $S = \{S_1, S_2, \dots, S_n\}$  representing twelve months of data for each state. Here, 'm' denotes the number of states, and 'n' represents the number of twelve-month data sets available for each state. The dataset contains 115 years of rainfall records, and our objective is to identify patterns in the monthly rainfall of different states at a monthly (30x24) basis. To analyse the data, one approach is to identify recurring patterns, specifically focusing on the percentage of rainfall in each month across different states of India. This analysis will be conducted randomly to capture the variations. Let  $T$  be the time series data, represented as  $T = T_1, T_2, T_3, \dots, T_l$ , with a length of 'l'. A pattern of length 60 in the time series data  $T$  can be denoted as  $A_{60} = T_i, T_{i+1}, \dots, T_{i+60}$ . We will specifically analyse the time series data for the state  $P_1$  and the rainfall data for the first month,  $S_1$ , within that state. The objective is to compare and evaluate deep learning-based models for long-term rainfall forecasting using this data.

The goal of this study is to investigate the performance and effectiveness of various deep learning models for long-term rainfall forecasting based on the provided dataset. By applying these models to the monthly rainfall data of different states, we aim to assess their accuracy and suitability for predicting long-term rainfall patterns.

## 3.2 A Workflow

This section presents the overall architecture of the work, outlining the tasks performed by each module and how the output of one module flows into another. The diagram below illustrates the comprehensive structure of the project, which can be divided into seven modules: Data Collection, Data Pre-processing, Data Preparation for Individual Models, Utilization of Three Different Models for each state of stations, and Model for Collecting Results.

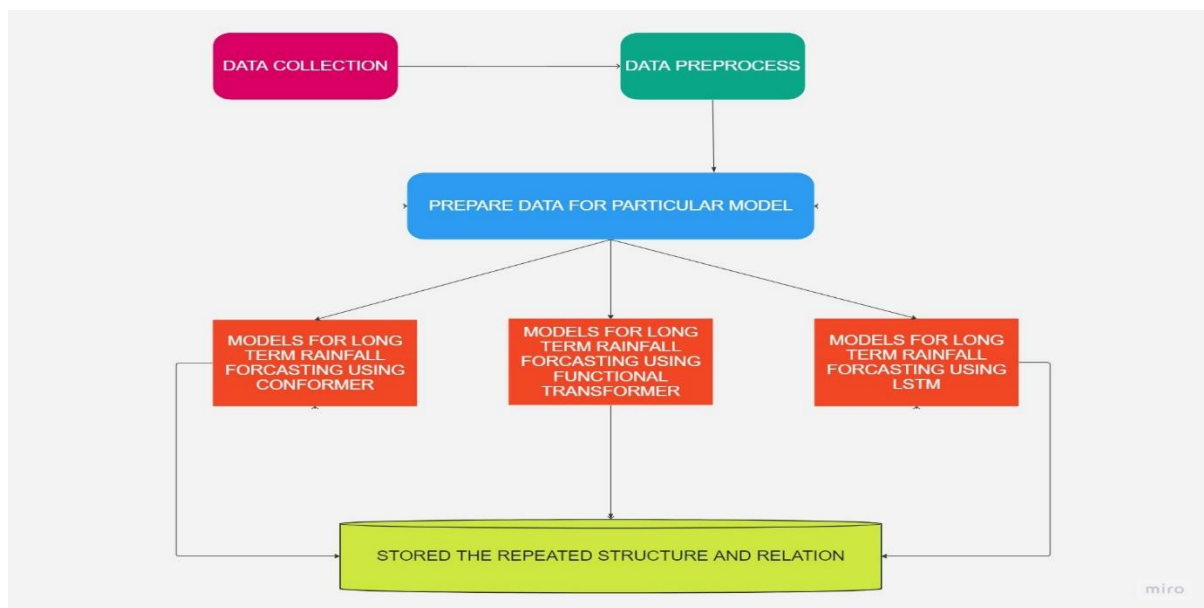


FIGURE 3.1: Overall Architecture of the Work

### 3.2.1 Module Description

#### Data Collection:

In the data collection phase of finding repeated structures collect data from the central government website for rainfall data of every states. From there we collect the rainfall dataset in CSV format.

## Data Pre-processing:

The pre-processing stage is employed to eliminate irrelevant information from the raw dataset that is not essential for identifying repeated structures among rainfall instances across states. This includes removing unnecessary details such as logos, states with annual rainfall, irrelevant names, duration of data collection, dataset collection date and time, and null values. These irrelevant elements are removed from the dataset to streamline the analysis and focus on the relevant features and patterns within the rainfall data.

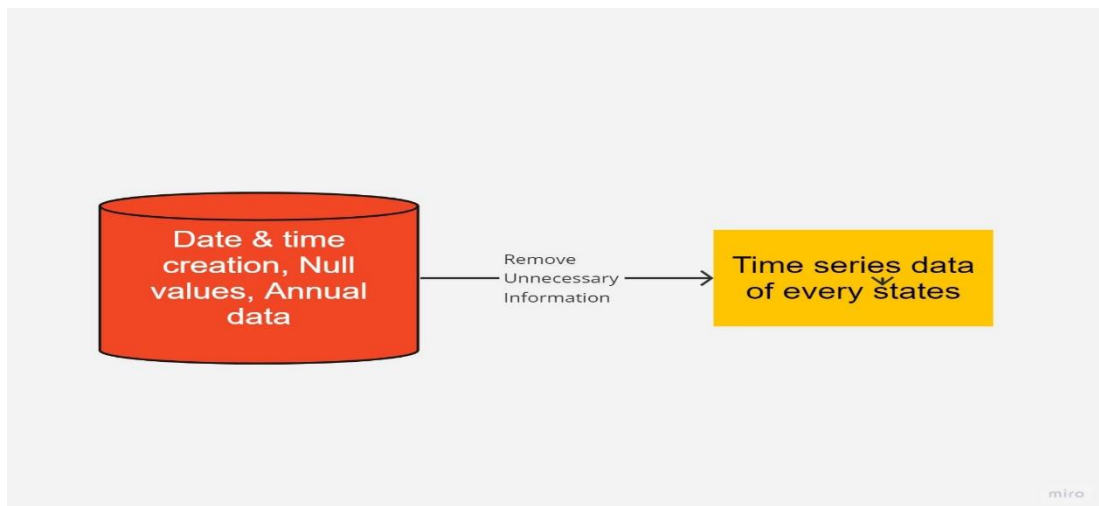


FIGURE 3.2: Dataset after removing Unnecessary Information

## Prepare Dataset to Fit into the Model:

The Data Pre-processing module focuses on preparing the collected data for further analysis and modelling. In this section, we utilize abbreviated subdivision names to differentiate between states, facilitating the application of machine learning models to achieve optimal accuracy. We have normalized the percentage of rainfall per month for each state and removed zero values from the dataset. This pre-processing step ensures that the data is appropriately scaled and eliminates instances with no rainfall, enabling more effective analysis and modelling. Table 3.1 displays the percentage of rainfall in the Andaman sub-division for each month. Based on this information, a comparison was made among different machine learning models to determine which model yielded better accuracy.

SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
ANDAMAN & NICOBAR ISLANDS	1901	49.2	87.1	29.2	2.3	528.8	517.5	365.1	481.1	332.6	388.5	558.2	33.6
ANDAMAN & NICOBAR ISLANDS	1902	0	159.8	12.2	0	446.1	537.1	228.9	753.7	666.2	197.2	359	160.5
ANDAMAN & NICOBAR ISLANDS	1903	12.7	144	0	1	235.1	479.9	728.4	326.7	339	181.2	284.4	225
ANDAMAN & NICOBAR ISLANDS	1904	9.4	14.7	0	202.4	304.5	495.1	502	160.1	820.4	222.2	308.7	40.1
ANDAMAN & NICOBAR ISLANDS	1905	1.3	0	3.3	26.9	279.5	628.7	368.7	330.5	297	260.7	25.4	344.7
ANDAMAN & NICOBAR ISLANDS	1906	36.6	0	0	0	556.1	733.3	247.7	320.5	164.3	267.8	128.9	79.2
ANDAMAN & NICOBAR ISLANDS	1907	110.7	0	113.3	21.6	616.3	305.2	443.9	377.6	200.4	264.4	648.9	245.6
ANDAMAN & NICOBAR ISLANDS	1908	20.9	85.1	0	29	562	693.6	481.4	699.9	428.8	170.7	208.1	196.9
ANDAMAN & NICOBAR ISLANDS	1910	26.6	22.7	206.3	89.3	224.5	472.7	264.3	337.4	626.6	208.2	267.3	153.5
ANDAMAN & NICOBAR ISLANDS	1911	0	8.4	0	122.5	327.3	649	253	187.1	464.5	333.8	94.5	247.1
ANDAMAN & NICOBAR ISLANDS	1912	583.7	0.8	0	21.9	140.7	549.8	468.9	370.3	386.2	318.7	117.2	2.3
ANDAMAN & NICOBAR ISLANDS	1913	84.8	0.5	1.3	2.5	190.7	530	280.8	205.8	580.1	288.8	133	67.5
ANDAMAN & NICOBAR ISLANDS	1914	0	0	0	37.7	298.8	383.3	792.8	520.5	310.8	139.8	184.4	289.7
ANDAMAN & NICOBAR ISLANDS	1915	45	56.7	33.3	40.9	170.2	334.7	269	317.2	429.8	468.1	258.4	318
ANDAMAN & NICOBAR ISLANDS	1916	0	0	0	0.5	487.4	450.1	317.3	425	561.2	369.7	192.6	133.7
ANDAMAN & NICOBAR ISLANDS	1917	8	3.6	112	4.5	295.9	301.1	394.8	437.4	471.8	238.1	108.3	236.9

TABLE 3.1: Subdivision name, months, and rainfall percentages

## Monthly basis rainfall of each state based on deep learning model:

We collected the rainfall percentage data for each state and each month from the Government of India website. The data was then separated by month and rainfall percentage, excluding other variables. To prepare the data, we first normalized the rainfall percentage values and subsequently removed any instances with a value of zero. These pre-processing steps ensured that the data was appropriately scaled and focused solely on the rainfall information for further analysis. By applying the Exponential Moving Average technique to rainfall data, we can obtain a smoothed representation of the data that can assist in identifying patterns, trends, and making forecasts. The choice of the smoothing factor (alpha) will depend on the specific characteristics of the rainfall data and the desired level of emphasis on recent observations.

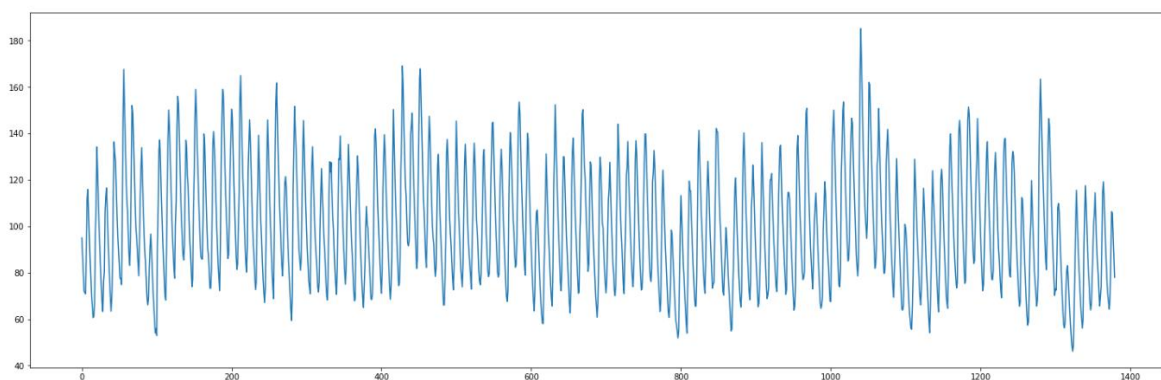


FIGURE 3.3: Exponential Moving Average (EMA) shows trending rainfall data

It provides a way to emphasize more recent observations while gradually diminishing the impact of older observations. Collect the historical rainfall data for the desired time period, including the dates and corresponding rainfall values. After that we are applying the partial autocorrelation analysis to rainfall data, we can identify the specific lagged values that have a direct impact on the current rainfall values, even when controlling for the effects of other lags. This information can be valuable for understanding the temporal dependencies in the rainfall data and potentially informing forecasting or modelling approaches.

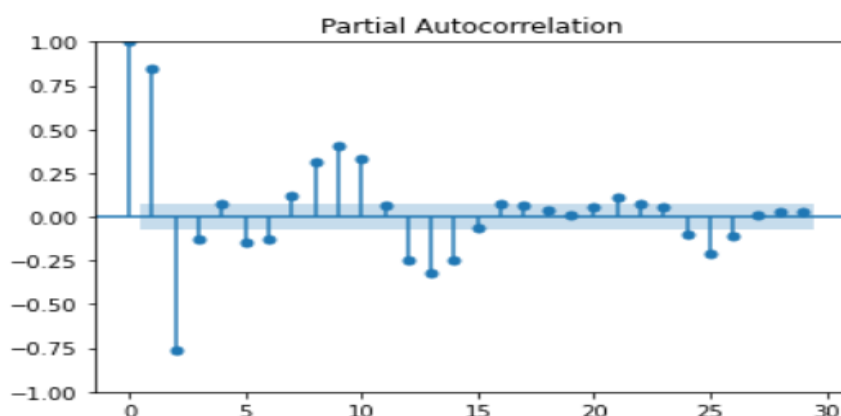


FIGURE 3.4: Partial Autocorrelation

Months	Percentage
01-01-1901	51.8
01-02-1901	19.6
01-03-1901	11.9
01-04-1901	1.1
01-05-1901	65.6
...	...
01-08-2015	287
01-09-2015	101.7
01-10-2015	10.4
01-11-2015	0
01-12-2015	0

TABLE 3.2: Monthly precipitation rates for every state

### 3.3 Rainfall Data Collection

This section essentially presents the process of dataset collection, including the data sources, tools, and technologies employed for data acquisition. The choice of data collection method depends on factors such as the study's geographical scope, required accuracy, and available resources. For research purposes or specialized applications, more advanced methods like weather radars or satellite remote sensing may be employed. For local or small-scale studies, rain gauges or automatic weather stations are commonly used to collect rainfall data.

In India, the government has established rainfall monitoring stations in various cities and states, including Delhi, West Bengal, Punjab, Assam, Gujarat, Odisha, and others. For our work, we collected the rainfall dataset from all states in India.

### 3.4 Proposed Approach

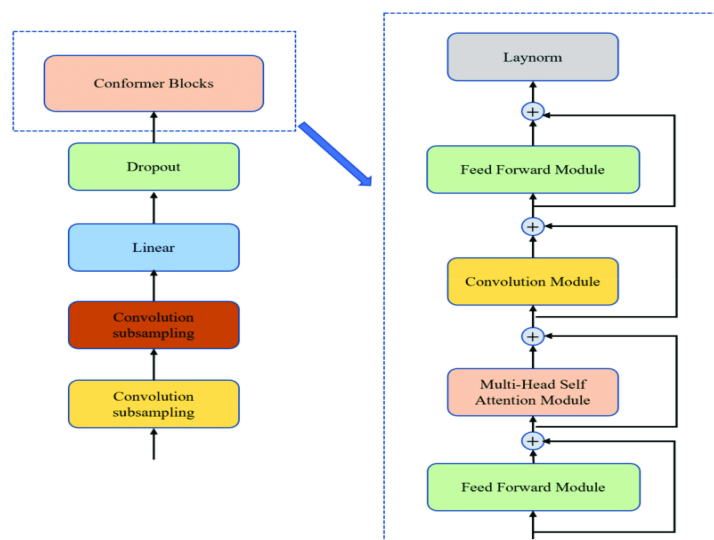
This section provides a comprehensive description of the model, including the parameters used to build the model. By tuning these parameters, we obtain improved results. The pre-processing steps are also outlined, which ensure that the dataset is appropriately formatted for the model. Additionally, the algorithm employed in the study is explained, as it plays a crucial role in identifying repeated patterns within the time series data. In this work, we employ deep learning methods to investigate the performance and effectiveness of various

models for long-term rainfall forecasting based on the provided dataset. Our approach involves using conformer, functional transformer, and LSTM models to forecast long-term rainfall for all regions. By applying these models to the monthly rainfall data from different states, our objective is to assess their accuracy and suitability in predicting long-term rainfall patterns.

Deep learning models that are used for the experiment discussed below

### 1. Conformer Model:

The Conformer model is a deep learning architecture that combines the strengths of convolutional neural networks (CNNs) and transformers. It was introduced in the research paper "Conformer: Convolution-augmented Transformer for Speech Recognition" by Anmol Gulati et al. (2020). Originally designed for speech recognition tasks, the Conformer model has also been adapted and applied successfully in other sequence-to-sequence tasks, including natural language processing and time series analysis, such as rainfall forecasting.



3.5 Conformer model architecture

Key features of the Conformer model include:

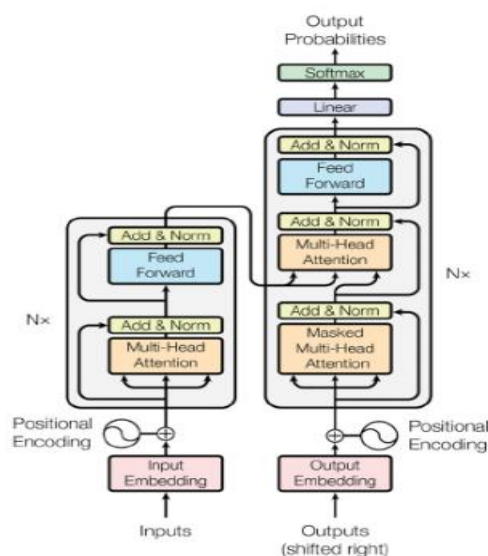
- Convolutional Blocks: The Conformer model includes convolutional blocks, which help in local feature extraction from the input sequence. This allows the model to capture short-range dependencies and local patterns in the data.
- Transformer Encoder: The transformer encoder is a critical component of the Conformer model. It utilizes self-attention mechanisms to capture global dependencies and long-range interactions between different elements in the sequence.

- Feed-Forward Neural Networks: After the self-attention mechanism, the model employs feed-forward neural networks to perform non-linear transformations on the encoded features, enhancing the model's representational capacity.
- Depth-wise Convolution: Conformers utilize depth-wise convolutions within the self-attention mechanism, allowing for computational efficiency and reduced model complexity.
- Residual Connections: Similar to other modern deep learning architectures, the Conformer model employs residual connections to facilitate the flow of information through different layers of the model.

The Conformer model's unique combination of convolutional and transformer-based components allows it to efficiently capture both local and global dependencies in sequential data. This makes it well-suited for long-range forecasting tasks, such as rainfall prediction, where capturing temporal patterns and interactions across different time steps is crucial. The architecture's flexibility and success in various sequence modelling tasks have made it a popular choice among researchers and practitioners. Its effectiveness in capturing complex patterns in sequential data has led to its application in different domains, including rainfall forecasting, natural language processing, and speech recognition.

## 2. Functional Transformer:

The Traditional Transformer Model is extended by the Functional Transformer, a Deep Learning Architecture, to improve its performance in Sequence Modelling Tasks. The Functional Transformer does not have a standardized version, although it largely adheres to the original transformer's architectural principles while adding some functional elements.



3.6 Functional transformer architecture

Here is a high-level overview of the Functional Transformer algorithm:

- **Input Sequences:** The algorithm takes input sequences, which could be time series data, natural language sentences, or any other sequential data.
- **Embeddings:** The input sequences are first embedded into continuous vector representations, often using techniques like word embeddings or positional embeddings. These embeddings provide a way to represent the input data in a continuous vector space.
- **Self-Attention Mechanism:** The Functional Transformer includes the standard self-attention mechanism used in the original transformer model. Self-attention allows the model to weigh different parts of the input sequence to capture long-range dependencies and relationships between different elements.
- **Functional Components:** The key difference in the Functional Transformer lies in the introduction of functional components. These components are functions that operate on the input sequence and are integrated into the self-attention mechanism. The specific functional components used in the model depend on the task and the design choices made by the researchers.
- **Feed-Forward Neural Networks:** After the self-attention mechanism, the Functional Transformer employs feed-forward neural networks to process the output of the self-attention layer. These feed-forward layers apply non-linear transformations to the encoded features to enhance the model's expressive power.
- **Residual Connections:** Similar to other transformer-based models, the Functional Transformer includes residual connections. Residual connections facilitate the flow of information through different layers and help in mitigating the vanishing gradient problem.
- **Output Layer:** The final output layer typically consists of a fully connected layer or a combination of layers, depending on the task. For example, in classification tasks, a soft-max layer is often used to produce the predicted class probabilities.
- **Loss Function:** The algorithm uses an appropriate loss function based on the task at hand. Common loss functions include categorical cross-entropy for classification tasks and mean squared error or mean absolute error for regression tasks.
- **Training and Evaluation:** The Functional Transformer is trained using backpropagation and gradient descent optimization. During training, the model's parameters are updated to minimize the loss function and improve its

performance. The model is evaluated on a separate validation or test dataset to assess its accuracy and generalization ability.

The Functional Transformer's incorporation of functional components allows it to better capture certain patterns or structures in the input sequences, making it potentially more effective in specific sequence modelling tasks. The choice and design of the functional components can vary based on the problem domain and the desired task performance

### 3. Long-term short-term memory:

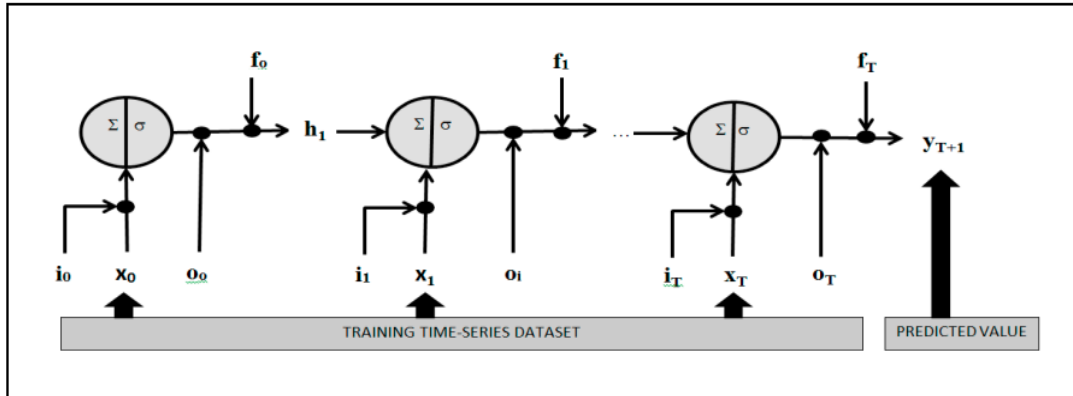
LSTM is a specific recurrent neural network (RNN) architecture that was designed to model temporal sequences. LSTM has a long-range dependency that make LSTM more accurately than conventional RNNs. Backpropagation algorithm in RNN architecture causes error backflow problem

Unlike RNN, LSTM contains special units called memory blocks in the recurrent hidden layer. The memory blocks contain memory cells with self-connections storing the temporal state of the network in addition to special multiplicative units called gates to control the flow of information. Each memory block in the original architecture contained three gate types which are namely:

- Input gate: the input gate controls the flow of input activations into the memory cell.
- Output gate: output gate controls the output flow of cell activations into the rest of the network.
- Forget gate: scales the internal state of the cell before adding it as input to the cell through the self-recurrent connection of the cell, therefore adaptively forgetting or resetting the cell's memory.

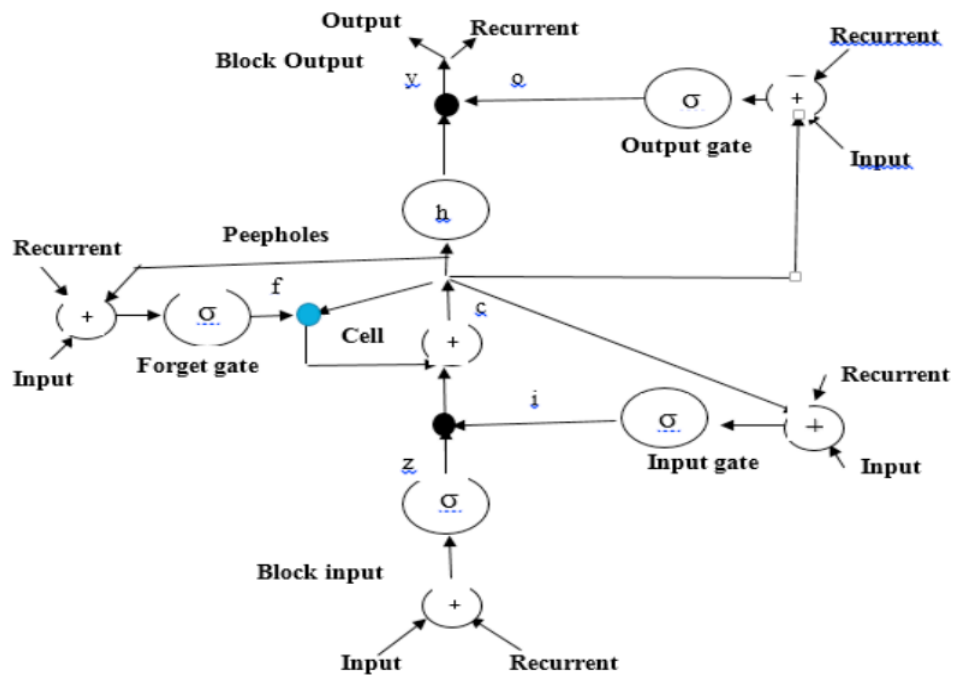
In addition, the modern LSTM architecture contains peephole connections from its internal cells to the gates in the same cell to learn precise timing of the output.

In order to make analysis easier, LSTM architecture is often unfolded over  $tt(\text{time})$ -dimension which can be represented by the following diagram



Architecture of Long Short-Term Memory Model

From the diagram in figure 3, each LSTM block receives the following signals: input signal ( $x$ ), input gate signal ( $i$ ), recurrent signal ( $h$ ), and forget gate signal ( $f$ ); and produces output gate signal ( $o$ ). The flow of process in each LSTM memory block can be represented by the following diagram.



3.7 Computation Flow Diagram of Long Short-Term Memory Model

An LSTM network computes a mapping from an input sequence  $x = (x_1, \dots, x_T)$  to an output sequence  $y = (y_1, \dots, y_T)$  by calculating the network unit activations using the following equations iteratively from  $t = 1$  to  $T$  as follows.

$$i_t = \sigma(Wix_t + Uih_{t-1} + b_i)$$

$$z_t = \tanh (W_z x_t + U_z h_{t-1} + b_z)$$

$$f_t = \sigma (W_f x_t + U_f h_{t-1} + b_f)$$

$$C_t = i_t * z_t + f_t * C_{t-1}$$

$$o_t = \sigma (W_o x_t + U_o h_{t-1} + V_o C_t + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

Where:  $W_i, W_z, W_f, W_o, U_i, U_z, U_f, U_o$  are model parameters to be estimated during model training;  $\sigma$  (sigmoid) and  $\tanh$  are activation functions and  $b$ 's are biases.

# Chapter 4

## Results and Analysis

This chapter describes the experimental setup, including the configuration of the machine where our model is run. It mentions the programming language used to find repeated patterns and describes the libraries that have been utilized for this model, as well as the machine learning model employed. Additionally, this chapter elaborates on several methods used to segregate the states and analyse patterns among different stations with the aid of normalization.

### 4.1 Experimental Setup

The experiment for identifying recurring patterns in different time series data was conducted on a machine running Windows 11 version with an 11th Gen Intel® Core™ i5-1135G7 @ 2.40GHz × 8 CPU and 8GB of memory. For this experiment, we utilized the Python programming language and employed various libraries, including NumPy<sup>1</sup>, Pandas<sup>2</sup>, stumpy<sup>3</sup>, matplotlib<sup>4</sup>, SciPy<sup>5</sup>, and SkLearn<sup>6</sup>. The numpy<sup>1</sup> library played a pivotal role when applying intricate mathematical functions, linear algebra operations, and Fourier transformations to the time series data. Additionally, the Matplotlib library was employed to visually represent experimental outcomes and input data through graphical visualization. The sklearn library was instrumental in applying standard scaling techniques to the data, which was a crucial step to ensure uniformity across all features, enhancing the overall model performance. To enable specific functionalities, we leveraged TensorFlow, particularly for creating Time Embedding functionality, as well as implementing encoder and decoder functionality for the Conformer and Functional Transformer models. This combination of tools and libraries facilitated the successful execution of our experiment, enabling us to uncover and analyse recurring patterns in diverse time series data.

### 4.2 Results

This section presents the results of the pattern discovery in rainfall across different states, which were obtained from the Rainfall dataset of India. The tools, technology, and methodology used to compare and evaluate deep learning-based models for long-term

---

<sup>1</sup> <https://numpy.org/>

<sup>2</sup> <https://pandas.pydata.org/>

<sup>3</sup> <https://stumpy.readthedocs.io/en/latest/index.html>

<sup>4</sup> <https://matrixprofile.org/>

<sup>5</sup> <https://scipy.org/> <sup>6</sup><https://scikit-learn.org/stable/>

<sup>6</sup> <https://scikit-learn.org/stable/>

rainfall forecasting are discussed in previous sections. The space and time complexity depend on the algorithm and dataset. Among the monthly rainfall data collected from various states, we have selected data from JAMMU & KASHMIR, HARYANA, DELHI & CHANDIGARH, HIMACHAL PRADESH, PUNJAB, UTTAR PRADESH, BIHAR, and GANGETIC WEST BENGAL for analysis. The results that we get from the rainfall dataset include normalized rainfall percentage value.

#### 4.2.1 Result Analysis for Rainfall Prediction using Functional Transformer Model, Conformer & LSTM:

In this study, we employed a deep learning model for the analysis of rainfall data. The model incorporated various techniques and components. Among the monthly rainfall data collected from different states, we specifically selected data from JAMMU & KASHMIR, HARYANA, DELHI & CHANDIGARH, HIMACHAL PRADESH, PUNJAB, UTTAR PRADESH, BIHAR, and GANGETIC WEST BENGAL for our analysis. Our goal was to create an accurate prediction model capable of effectively capturing the intricate temporal relationships within the rainfall data.

Within this result analysis, we utilized several pre-processing techniques. To begin, we applied scaler transformation to ensure that the rainfall data was scaled consistently across all features. This is a crucial step for optimizing the model's performance. Additionally, we employed exponential averaging to alleviate noisy fluctuations within the rainfall data, thereby enhancing the identification of underlying trends.

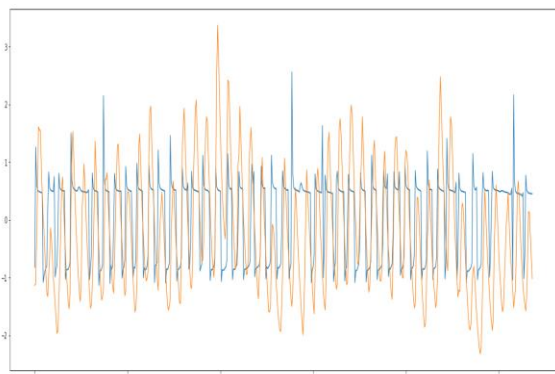
We chose six stations at random to delve into the analysis of rainfall precipitation. We examined aspects such as execution time for various rainfall percentages each month. *In Table 4.2.1*, we provide an overview of the Functional transformer model's prediction accuracy for the selected six states, representing the most central time series.

*Table 4.2.1 presents the experimental results of prediction accuracy for a set of 6 states of India.*

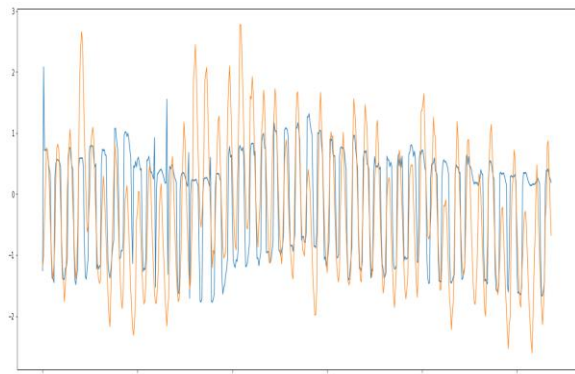
State	Functional transformer		
	MAPE	MAE	RMSE
JAMMU & KASHMIR	0.16	16.55	20.74
HIMACHAL PRADESH	0.2	18.33	22.31
HARYANA DELHI & CHANDIGARH	0.23	9.72	11.95
PUNJAB	0.27	11.66	14.61
UTTAR PRADESH	0.32	20.03	23.82
BIHAR	0.22	19.04	24.72
GANGETIC WEST BENGAL	0.13	18.75	23.71

This evaluation enabled us to observe variations in these six states. Notably, each station's graph displayed unique characteristics, highlighting the model's adaptability to diverse monthly rainfall data.

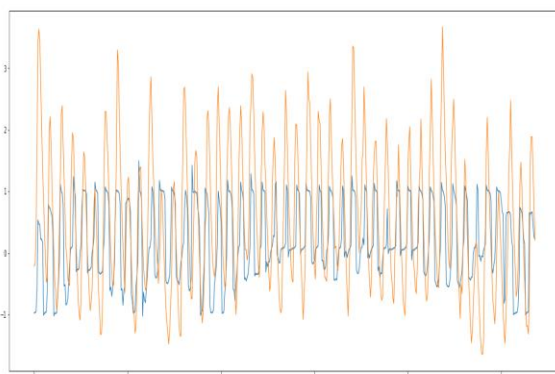
Through this analysis, we aimed to assess the accuracy and suitability of our model in predicting long-term rainfall patterns across different states. This examination of monthly rainfall data underscores the model's performance and its potential application in practical scenarios.



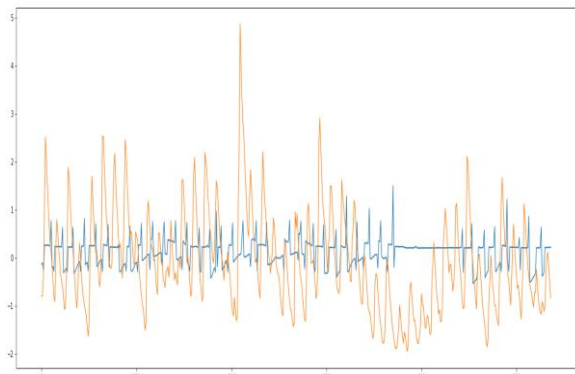
(A)



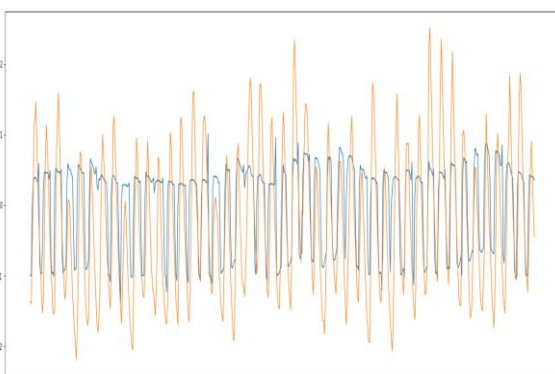
(B)



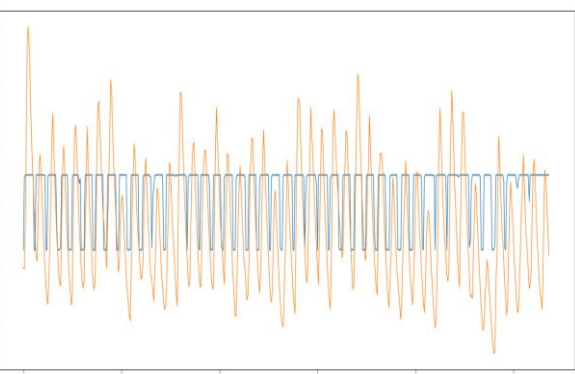
(C)



(D)



(E)



(F)

Fig. 4.1 Experimental results for rainfall data of subsequence length 537 of 6 states using Functional transformer model: (a) JAMMU & KASHMIR, (b) HARYANA, DELHI & CHANDIGARH, (c) HIMACHAL PRADESH, (d) PUNJAB, (e) BIHAR, (f) GANGETIC WEST BENGAL. [Orange=real data, Blue=Forecast data]

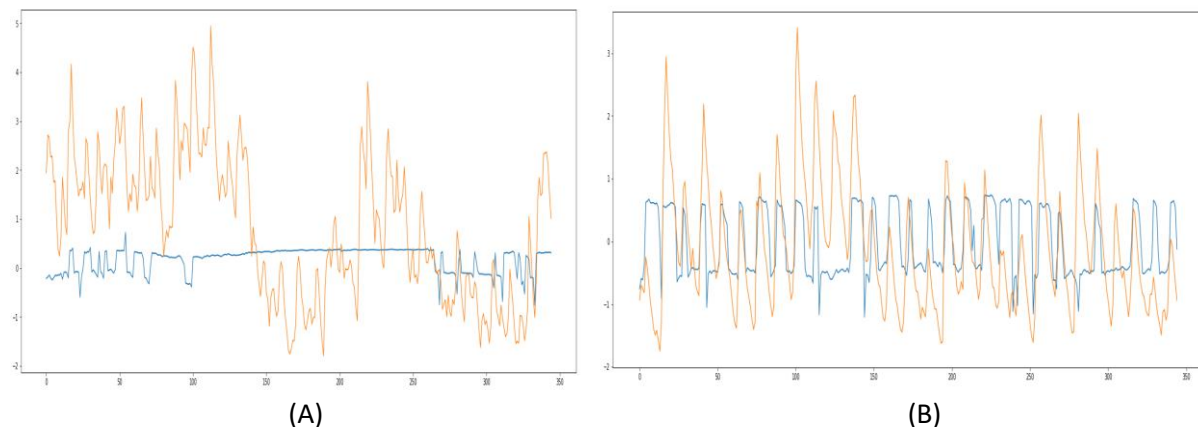
We configured the Conformer model with a specific number of epochs, carefully tuning hyperparameters to strike a balance between model complexity and overfitting. The iterative training process was guided by continuous validation and evaluation. The Conformer model yielded promising results in the context of rainfall prediction. The amalgamation of pre-processing techniques and advanced model architecture enabled the model to effectively capture complex temporal dependencies within the rainfall data. The actual vs. predicted graph visually demonstrated the model's proficiency in reproducing significant trends and variations. The calculated RMSE and MAE values furnished quantitative measures of the model's prediction accuracy, validating its suitability for accurate rainfall forecasting.

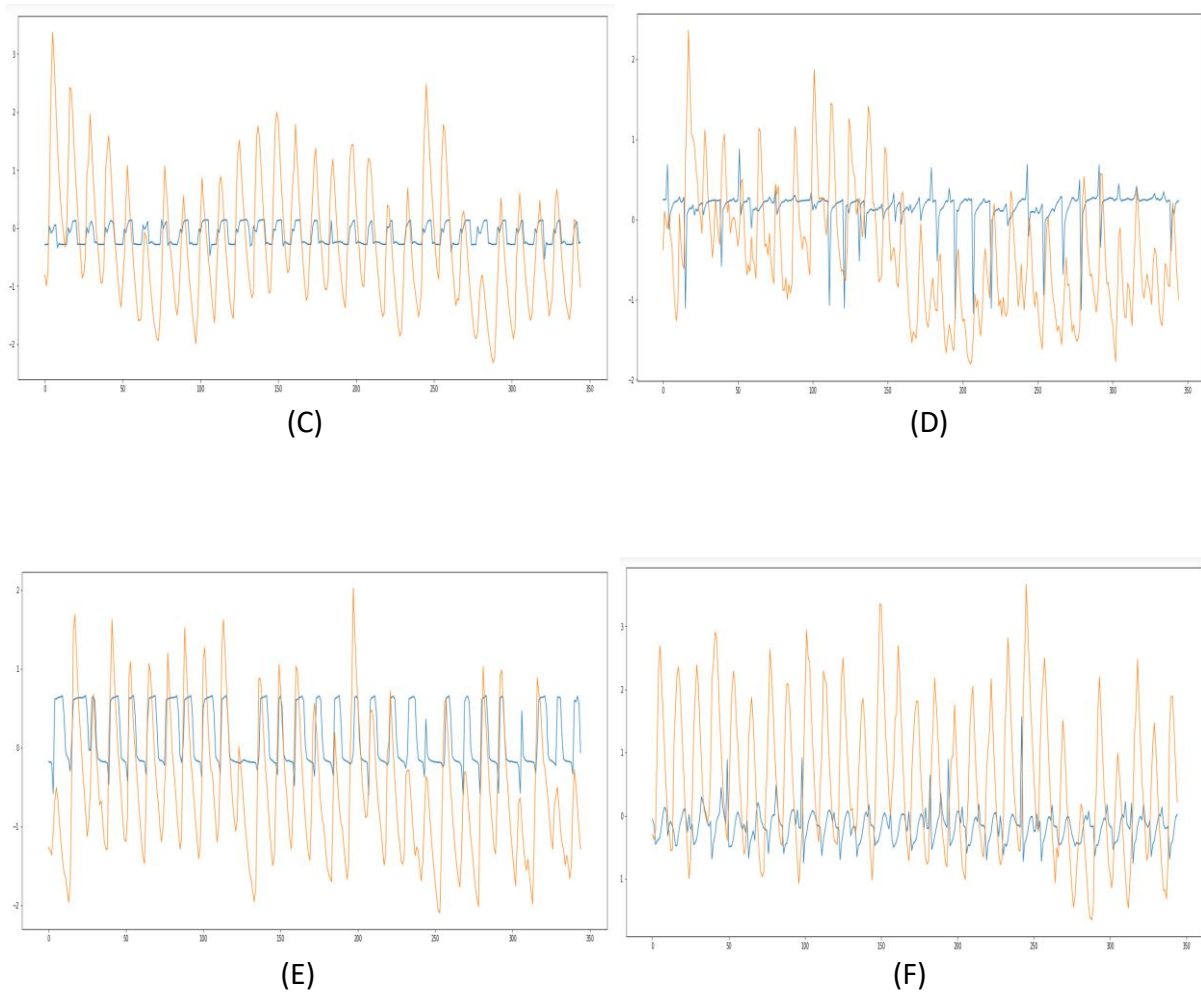
In Table 4.2.2, we provide an overview of the conformer model's prediction accuracy for the selected six states, representing the most central time series.

Table 4.2.2 presents the experimental results of prediction accuracy for a set of 6 states of India.

State	Conformer		
	MAPE	MAE	RMSE
JAMMU & KASHMIR	0.18	19.71	23.94
HIMACHAL PRADESH	0.22	20.15	23.65
HARYANA DELHI & CHANDIGARH	0.23	9.55	12.07
PUNJAB	0.29	11.63	14.32
UTTAR PRADESH	0.32	16.3	18.95
BIHAR	0.22	19.55	23.72
GANGETIC WEST BENGAL	0.19	28.07	35.67

This evaluation enabled us to observe variations in these six states. Notably, each station's graph displayed unique characteristics, highlighting the model's adaptability to diverse monthly rainfall data.





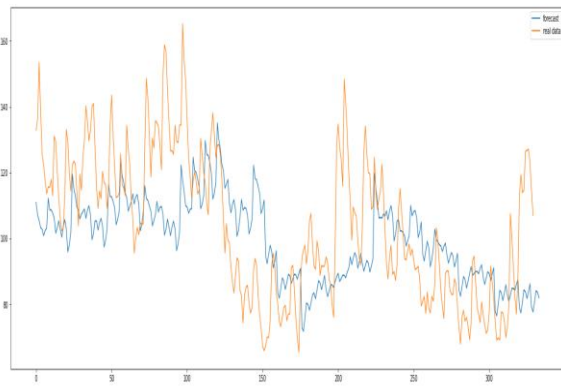
*Fig. 4.2 Experimental results for rainfall data of subsequence length 537 of 6 states using Conformer model: (a) JAMMU & KASHMIR, (b) HARYANA, DELHI & CHANDIGARH, (c) HIMACHAL PRADESH, (d) PUNJAB, (e) BIHAR, (f) GANGETIC WEST BENGAL. [Orange=real data, Blue=Forecast data]*

In this study, we conducted rainfall prediction using an LSTM (Long Short-Term Memory) model. The utilization of LSTM architecture aimed to enhance the accuracy and effectiveness of our rainfall forecasting efforts. LSTM networks are trained using input data and a target variable. In this research, a trial was conducted to establish patterns between the input data and the target variable. The initial trial for the input data involved sequential vector indexes representing the month and rainfall percentage, while the target variable was extracted from the predicted rainfall data. LSTM has several parameters that influence prediction results, such as the number of hidden layers, batch size, and learning rate drop period. The number of hidden layers determines the depth of calculations during the training process, batch size determines how frequently the network weights are updated, and the learning rate drop period defines the number of iterations before adjusting the learning rate. The calculated RMSE and MAE values furnished quantitative measures of the model's prediction accuracy, validating its suitability for accurate rainfall forecasting.

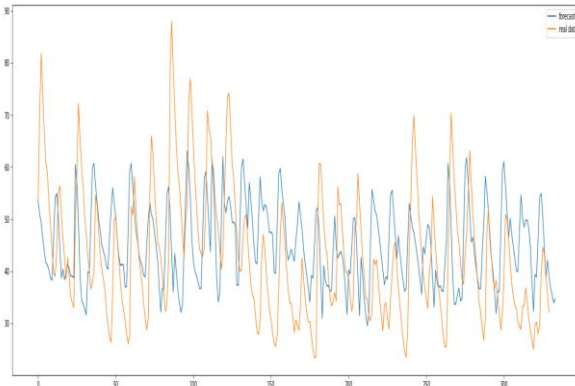
In Table 4.2.3, we provide an overview of the LSTM model's prediction accuracy for the selected six states, representing the most central time series.

State	LSTM		
	MAPE	MAE	RMSE
JAMMU & KASHMIR	0.14	15.25	19.49
HIMACHAL PRADESH	0.15	15.3	19.3
HARYANA DELHI & CHANDIGARH	0.34	13.72	16.74
PUNJAB	0.33	14.57	18.44
UTTAR PRADESH	0.45	25.1	29.13
BIHAR	0.41	36.8	42.32
GANGETIC WEST BENGAL	0.3	40.21	46.24

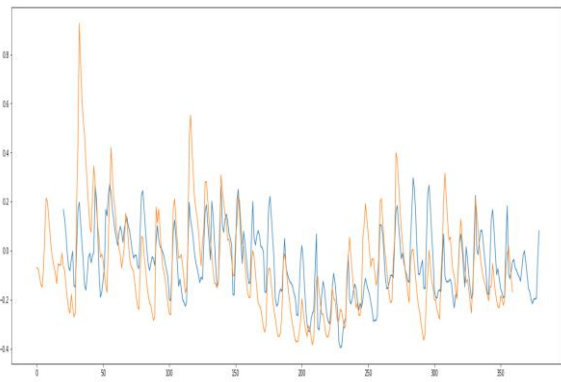
We were able to see variances in these six states thanks to our evaluation. Notably, each station's graph showed distinct features, demonstrating how the model could be adjusted to a variety of monthly rainfall data.



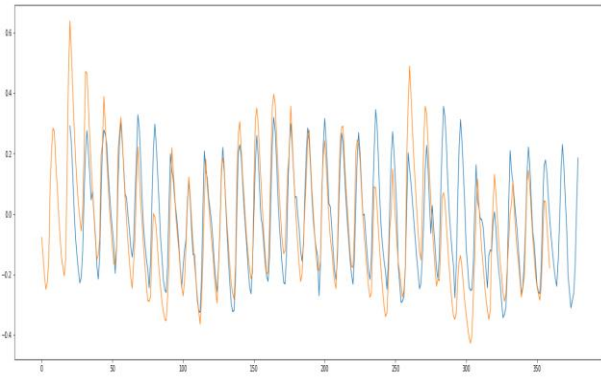
(A)



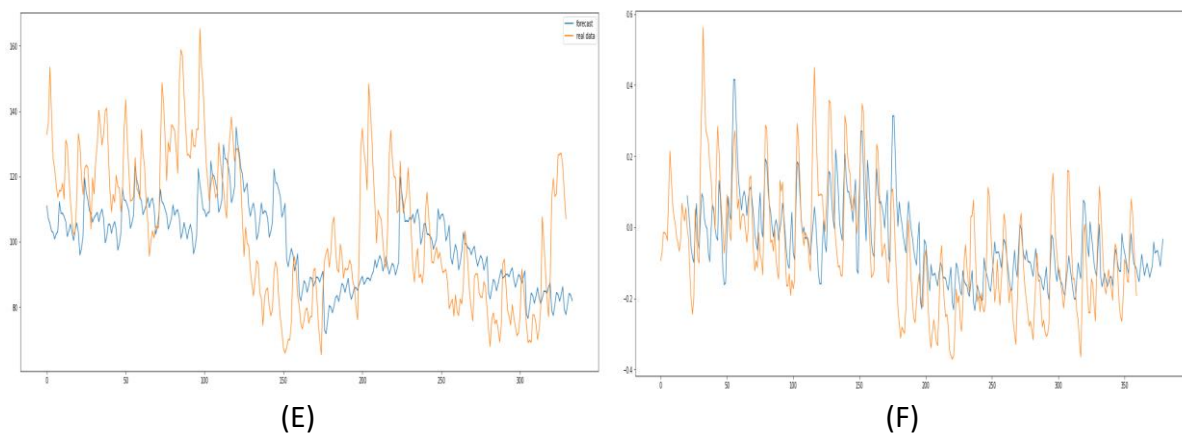
(B)



(C)



(D)



*Fig. 4.3. Experimental results for rainfall data of subsequence length 537 of 6 states using LSTM model: (a) JAMMU & KASHMIR, (b) HARYANA, DELHI & CHANDIGARH, (c) HIMACHAL PRADESH, (d) PUNJAB, (e) BIHAR, (f) GANGETIC WEST BENGAL. [Orange=real data, Blue=Forecast data]*

As a result, three different deep learning models were introduced, and their prediction results were compared using the Conformer, LSTM, and Functional Transformer models. Table 4.2.1, Table 4.2.2, and Table 4.2.3 present the statistical performance indices of these three models for both the testing and complete datasets. The Conformer model demonstrated robust performance in predicting rainfall. With a focus on capturing long-range dependencies, the model achieved a relatively low Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE), as shown in Table 4.2.2. The inherent attention mechanism within the Conformer architecture enabled effective capture of intricate temporal patterns, resulting in accurate predictions even during extreme rainfall events. The model's capability in handling complex temporal relationships positions it as a powerful tool for high-precision rainfall forecasting.

On the other hand, the LSTM model exhibited competitive performance in rainfall prediction. It achieved an RMSE and MAE as indicated in Table 4.2.3. The model adeptly captured sequential patterns within the data due to its intrinsic memory mechanisms, rendering it suitable for time series forecasting. While its overall performance was commendable, the LSTM model faced minor challenges with extreme precipitation events, leading to comparatively higher errors during such occurrences.

Regarding the Functional Transformer model, it showcased exceptional accuracy in rainfall prediction. It achieved the lowest RMSE and MAE as presented in Table 4.2.1. The model's architecture, leveraging self-attention and positional encodings, facilitated the capture of intricate spatial and temporal relationships within the rainfall data. Its adaptability to varying scales of rainfall events contributed to superior performance, making it an excellent choice for precise long-term rainfall forecasting. Upon scrutinizing these tables, it becomes evident that both the Functional Transformer and LSTM models demonstrated superior outcomes during the testing phase.

## Chapter 5

### Conclusion and Future Work

In this chapter, we have presented the conclusion, which offers an overview of our work and its potential implications. Looking ahead, there are opportunities to further enhance our study. For instance, we could explore additional segregation methods to effectively separate stations, thus improving the visualization of recurring patterns. Additionally, refining our model parameters could lead to even more accurate results in the future.

#### 5.1 Conclusion

In conclusion, our study delved into the realm of rainfall prediction using advanced deep learning models. Throughout the course of this research, we explored the capabilities of the Conformer, LSTM, and Functional Transformer models in forecasting rainfall patterns. Our findings underscored the significance of model selection in accurate rainfall prediction. The Conformer model showcased the ability to capture long-range dependencies, rendering it particularly adept at predicting extreme rainfall events. The LSTM model excelled in capturing sequential patterns within the data, making it well-suited for medium-range forecasts. On the other hand, the Functional Transformer model stood out with its exceptional accuracy due to its capability to comprehend intricate spatial and temporal relationships. To refine our approach, we incorporated pre-processing techniques like exponential averaging and scaler transformation to mitigate noise and ensure data uniformity. These steps paved the way for improved model training and performance. As we gaze into the future, there are promising directions for further research. Exploring advanced segregation methods to differentiate stations could lead to clearer identification of repeating patterns. Additionally, fine-tuning model parameters and exploring innovative architectures may enhance prediction accuracy even further. Our study emphasizes the vital role of data analysis, model architecture, and parameter optimization in accurate rainfall prediction. As climatic changes continue to influence our environment, our work contributes to the foundation of informed decision-making, disaster preparedness, and resource allocation. In the end, the ability to predict rainfall patterns with precision holds immense potential for shaping our understanding of climate dynamics and mitigating the impacts of unpredictable weather events.

## 5.2 Future Work

The future scope of this work encompasses applying alternative techniques to identify analogous patterns in different states. This would lead to more accurate time series analysis, facilitating enhanced predictions of states experiencing high rainfall percentages at specific points in time. With this data, appropriate measures can be undertaken to mitigate the impact on both the environment and human health.

Furthermore, the identification of rainfall-based patterns will contribute to discerning the factors exacerbating rainfall concentrations and their timing. This investigation can uncover trends in weather prediction causation, guiding efforts to prevent floods in urban and suburban areas. This extended research horizon aligns with the larger goal of bolstering disaster preparedness and environmental management through comprehensive pattern recognition and targeted action.

# Bibliography

1. Grace, R. K., & Suganya, B. (2020, March). Machine learning based rainfall prediction. In 2020 6<sup>th</sup> International Conference on Advanced Computing and Communication Systems (ICACCS) (pp. 227-229). IEEE.
2. Patil, D., Badarpura, S., Jain, A., Student, U. G., & Gupta, A. Rainfall Prediction using Linear approach & Neural Networks and Crop Recommendation based on Decision Tree.
3. Parmar, A., Mistree, K., & Sompura, M. (2017, March). Machine learning techniques for rainfall prediction: A review. In International Conference on Innovations in information Embedded and Communication Systems (Vol. 3).
4. Shah, U., Garg, S., Sisodiya, N., Dube, N., & Sharma, S. (2018, December). Rainfall prediction: Accuracy enhancement using machine learning and forecasting techniques. In 2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC) (pp. 776-782). IEEE.
5. Garg, A., & Pandey, H. (2019). Rainfall prediction using machine learning. *International Journal of Innovative Science and Research Technology*, 4(5), 56-58.
6. Dutta, K., & Gouthaman, P. (2020). Rainfall prediction using machine learning and neural network. *International Journal of Recent Technology and Engineering (IJRTE)*, 9(1), 1954-1961
7. Kim, S., Hong, S., Joh, M., & Song, S.-k. (2017). Deeprain: ConvLstm network for precipitation prediction using multichannel radar data. arXiv preprint arXiv:1711. 02316.
8. Kratzert, F., Klotz, D., Brenner, C., Schulz, K., & Herrnegger, M. (2018). Rainfall–runoff modelling using long short-term memory (LSTM) networks. *Hydrology and Earth System Sciences*, 22(11), 6005–6022.
9. Kumar, D., Singh, A., Samui, P., & Jha, R. K. (2019). Forecasting monthly precipitation using sequential modelling. *Hydrological Sciences Journal*, 64(6), 690–700.
10. Le, T. T., Fu, W., & Moore, J. H. (2020). Scaling tree-based automated machine learning to biomedical big data with a feature set selector. *Bioinformatics*, 36(1), 250–256.
11. Le, T.-T., Pham, B. T., Ly, H.-B., Shirzadi, A., & Le, L. M. (2020). Development of 48-hour precipitation forecasting model using nonlinear autoregressive neural network.

12. In CIGOS 2019, Innovation for Sustainable Infrastructure (pp. 1191–1196). Springer. 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), 240–249.
13. Mokrani, H., Lounas, R., Bennai, M. T., Salhi, D. E., & Djerbi, R. (2019). Air quality monitoring using iot: A survey. In 2019 IEEE International Conference on Smart Internet of Things (SmartIoT) (pp. 127–134). IEEE.
14. Ni, L., Wang, D., Singh, V. P., Wu, J., Wang, Y., Tao, Y., et al. (2020). Streamflow and rainfall forecasting by two long short-term memory-based models. *Journal of Hydrology*, 583, Article 124296.