Deep Learning Based Short-range Forecasting of Indian Summer Monsoon Rainfall using Earth Observation and Ground Station Datasets

Bipin Kumar[a], Namit Abhishek[b], Rajib Chattopadhyay[a,e], Sandeep George[c], Bhupendra Bahadur Singh[a], Arya Samanta[b], B.S.V. Patnaik[c], Sukhpal Singh Gill[f] Ravi S. Nanjundiah[d] and Manmeet Singh[a]

[a]Indian Institute of Tropical Meteorology, Ministry of Earth Sciences, Dr. Homi Bhabha Road, Pashan, Pune, 411008, India; [b]Indian Institute of Science Education and Research, Dr. Homi Bhabha Road, Pune, 411008, India; [c]Indian Institute of Technology Madras, Chennai, 600036, India; [d]Center for Atmospheric and Ocean Sciences, Indian Institute of Science, Bengaluru, 560012, India; [e]CRS, India Meteorological Department, Pune, 411001, India; [f]Queen Mary University of London Mile End Road, London E1 4NS, UK.

ARTICLE HISTORY

Compiled September 7, 2022

ABSTRACT

We develop a deep learning model (DL) for Indian Summer Monsoon (ISM) short-range precipitation forecasting using a ConvLSTM network. The model is built using daily precipitation records from both ground-based observations and remote sensing. Precipitation datasets from the Tropical Rainfall Measuring Mission and the India Meteorological Department are used for training, testing, forecasting, and comparison. For lead days 1 and 2, the correlation coefficient (CC), which was determined using predicted data from the previous five years and corresponding observational records (from both in-situ and remote sensing products), yielded values of 0.67 and 0.42, respectively. Interestingly, the CCs are even higher over the Western Ghats and Monsoon trough region. The model performance evaluated based on skill scores,Normalized Root Mean Squared Error (NRMSE), Mean absolute percentage error (MAPE) and ROC curves show a reasonable skill in short-range precipitation forecasting. Incorporating multivariable-based DL has the potential to match or even better the forecasts made by the state-of-the-art numerical weather prediction models.

KEYWORDS

ConvLSTM model; Remote sensing ; TRMM Data; Station Data; Indian Summer Monsoon ; Short-range forecasting; Custom Loss Function.

1. Introduction

The modeling studies on monsoon have been traditionally performed using numerical models of the weather and climate (Krishnan et al. 2020), which solve partial differential equations of the atmosphere-ocean-land coupled systems. In general, methods for predicting different meteorological variables use numerical weather prediction (NWP)

CONTACT Bipin Kumar Email: bipink@tropmet.res.in

techniques by solving a set of higher-order non-linear differential equations. In the Indian context, the models focusing on different temporal scales of the Indian monsoon, viz., short-range (1-3 days) to climate scale (10's years), are being used in research and operational mode to understand the monsoon better and disseminate the information to the stakeholders. In recent times, the need for better forecasting has risen for several specific applications whereas the skills of dynamical models are still modest. Numerical models have their own limitations, which is being further complicated by additional forcing in the form of climate change.

Short-range weather forecasting, is significant, particularly, in the context of the monsoon as high-impact weather events are increasing with global warming (Goswami et al. 2006; Pörtner et al. 2022). An accurate assessment of sub-district scale weather a few days ahead can arm the administration to take necessary measures. For example, the usage of NWP towards short-range precipitation forecasts can help in mitigating the impacts of cloud bursts, and heavy-to-very-heavy extreme rainfall etc. Short-range weather prediction in India is being carried out by a suite of dynamical models; at present the highest spatial resolution of these models in operational mode is ≈12.5km (Rajeevan and Santos 2020; Mukhopadhyay et al. 2019). Having shown tremendous progress in the last decade, such models sometimes fail to capture extreme rainfall events and/or do not produce realistic rains on the land regions. For example, the National Centers for Environmental Prediction (NCEP) based Global Forecast System (GFS) T1534 (≈12.5 km), has shown significant improvements in the short-range operational forecasts over India (Mukhopadhyay et al. 2019). However, even such an advanced model underestimates heavy to very heavy rainfall, while the extremely heavy rainfall categories are only better at shorter lead times. There could be various reasons for these issues, such as the complex, non-linear and turbulent weather in the tropical regions and the usage of parameterization schemes generating precipitation in the model. When the forecasts are issued in ensemble mode, one more issue is the uncertainty or spread associated with such forecasts which need to be properly taken care of. Some recent studies have suggested a mixed approach to improve the existing forecasts, which we would discuss in the subsequent text.

Other than numerical models, statistical and feature selection-based Artificial Neural Network (ANN) models have been used in the past to predict rainfall at different time scales with some success (Saha et al. 2016; Dasgupta et al. 2020). These models employ two popular concepts: feature selection and followed by prediction using statistical or simple machine learning algorithms (Saha et al. 2016; Goswami and Xavier 2003; Chattopadhyay et al. 2008). Recently, machine learning algorithms have also been applied to generate rainfall forecasts at short and long time scales (Moon et al. 2019; Diez-Sierra and del Jesus 2020). Yin et al. (2022) combined support vector machine (SVM) regression with quantile-based bias correction method to improve real-time NWP based precipitation forecasts. Moghaddam et al. (2022) found that use of ML or DL based inference with numerical models improves surface/ground fluxes beneath river systems. Ehsani et al. (2021) applied ML-based retrieval algorithm to obtain statistically better estimation of snowfall than remote sensing based datasets alone. Samadianfard et al. (2022) reported that implementing classification algorithms and decision trees along with past meteorological data, improves prediction of daily precipitation. Recently, ensemble weather forecasting has picked up pace instead of relying on a single NWP model output. Again, it is important to incorporate the uncertainty in different ensemble members. Bias and dispersion errors in NWP ensemble precipitation forecasts are minimised through statistical post-processing. However, traditional approaches only incorporate ensemble mean as the predictor, ignoring ensemble spread which can serve as a crucial parameter for forecast uncertainty. Zhao et al. (2022) proposed forecasts calibrated using the two-step calibration considering both the dynamically flow-dependent ensemble spread from raw ensemble forecasts and the seasonally coherent calibrated ensemble spread that contains statistically generated uncertainty information. The study reported an improvement in the forecasts as compared to those done only with seasonally coherent calibrations.

In the last decade, deep learning has emerged as a potential methodology to solve complex, non-linear problems by un-wrapping the nonlinearities in different layers of the deep neural network (Zeiler and Fergus 2014). Moghaddam et al. (2021) applied a deep learning model, UNET, to determine the effective hydraulic conductivity and reported the ML/DL techniques can be applied to other areas of surface hydrology. The non-linear operators that have gained prominence in the Computer Vision community can be applied to weather and climate science problems, particularly the problem of deciphering accurate precipitation forecasts in the numerical weather prediction models (Reichstein et al. 2019). For example, forecasts based on NWP models suffer from biases which need to be corrected while issuing forecasts. Li et al. (2022) developed a convolutional neural network (CNN)-based post-processing method for precipitation forecasts which outperformed traditional methods in forecast accuracy and reliability, especially for heavy rain events. Recent studies prove that progress made by dynamic models should not be ignored in favour of deep learning, but rather should be supplemented by emerging new techniques (Dasgupta et al. 2020; Singh et al. 2021b).

Deep learning methods can learn complex mapping between inputs and outputs which result in better forecasts. These methods have shown remarkable results in various fields including meteorology, where it can be used to forecast the precipitation (Shi et al. 2017). The present study aims to develop a deep learning model for forecasting spatio-temporal sequences and apply it to ISM precipitation data obtained from satellite and ground-based stations. Satellite based precipitation estimates are obtained from the Tropical Rainfall Measuring Mission (TRMM) (see Huffman et al. (2010)) data have been used to understand precipitation processes in several studies (see Singh et al. (2021a) and references therein). Also, we use gridded data prepared by the India Meteorological Department (IMD) which provides long-term reliable precipitation data based on in-situ ground station records. Recently these satellite and ground-based rainfall estimates are being widely used to develop empirical as well machine learning models.

Viswanath et al. (2019) attempted to study the active and break spell of monsoon using Long Short Term Method (LSTM)-based networks. Borah et al. (2013) used self organizing maps (SOM) for the ensemble extended range forecast of active/break cycles ISM. In the work by Barzegar et al. (2021), a CNN-LSTM based model was employed to forecast the level of water in a lake. Li et al. (2022) used a CNN-LSTM-based deep learning method to predict 3-hour precipitation, which outperformed traditional machine learning methods in terms of prediction performance. The study by Siami-Namini et al. (2018) also shows the power of LSTM which outperformed the Autoregressive Integrated Moving Average (ARIMA) model by reducing the error rate up to 87%. Further studies, such as (Khan and Maity 2020) , have shown the effectiveness of using a hybrid model with conv2D and Multi-Layer Perceptron (MLP) to do a multivariate prediction for rainfall. When compared with a simple MLP and an SVM, this hybrid model was found better. The convolutional 1D and MLP together better captured the complex relationship of rainfall with the other variables. The work by researchers in (Ham et al. 2019; Saha and Nanjundiah 2020) demonstrates the power of convolutional neural-network-based architecture to predict the El Nino–Southern Oscillation (ENSO) variations effectively. Their model was able to give skillful forecasts for lead time up to one and a half years. The nino3.4 index of the model was found to be better than other state-of-the-art dynamic models. Most of the above models used either only convolutional or LSTM based architectures to capture rainfall patterns. These models also only tried to either classify or detect patterns in the future.

For precipitation forecasting, a model has to be more powerful, able to capture the temporal and spatial structure of the data and hence, we note the usage of ConvLSTM based architectures in (Shi et al. 2017; Kim et al. 2017; Shi et al. 2015). In Shi et al. (2015), the effectiveness of ConvLSTM over linear regression is established by working with multichannel radar data. ConvLSTM model is a hybrid model that uses spatio-temporal information to generate the forecast. For dispersive waves (such as Rossby waves, convectively coupled waves) which have typical wave-frequency spectral signatures and generate skewed weather states (e.g. extreme weather events), this type of spatio-temporal information based model is a natural choice. For the present study, we, therefore, chose this model as we want a state-of-art model (Shi et al. 2015) which is already successful in similar applications but has not been applied for the monsoon forecast. The research by Shi et al. (2015) is the best work for the application of ConvLSTM, where the model's effectiveness is established for spatio-temporal sequence prediction problems. The ConvLSTM based model was also shown to outperform the state-of-the-art optical flow-based ROVER algorithm (Shi et al. 2015). The above points motivate us to utilize it for precipitation forecasting in this study. A sketch of the network used in this work, based on Shi et al. (2015), is shown in Figure 1. The main contribution of this work is the application of a hybrid AI models for precipitation forecasting. We evaluated three models, ARIMA and ConvGRU and ConvLSTM. The ConvLSTM approach was chosen for this work. The conclusions drawn from these methods are discussed. After being integrated with a high-performance computing application, it is anticipated that the ConvLSTM will be extremely efficient (with a large training cycle, for example). Our prototype model is only a starting point for further development. As mentioned before, in this study, we have worked on two types of Geoscience data for forecasting precipitation. One of them is the ground-based in-situ precipitation data from the India Meteorological Department (IMD) and the other is remotely-sensed Tropical Rainfall Measuring Mission (TRMM) data which includes data from (i) Lightning Imaging Sensor (ii) TRMM Microwave Imager, and (iii) Visible Infrared Scanner.

The next section provides details of the data and methodology used in this study. The problem statement is described in Section III and the architecture of Artificial Intelligence (AI) model developed for sub-district (25× 25km) scale and aimed towards short range (1-3 days) forecasting is explained in section IV. Section V provides descriptions and discussion on the results obtained from the model. This study's conclusions are contained in section VI. The possible future work is provided in the last section.

2. METHODOLOGY AND DATA

The LSTM networks were first introduced by Hochreiter and Schmidhuber (1997). It typically has a forget gate, an input gate, an output gate with its weights in which it can control what information to retain and what to forget, thus learning long-term associations. **?** developed the architecture of ConvLSTM when designing a model for learning spatio-temporal correlation in precipitation nowcasting problem. In this

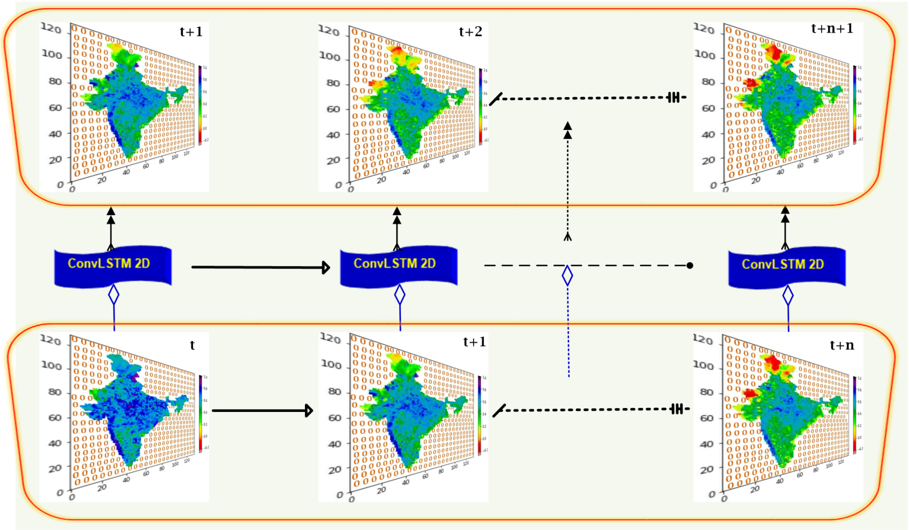


Figure 1. A sketch of ConvLSTM architecture based on (Shi et al., 2015) used in this study.

architecture, convolutional operations replace the typical fully connected architecture within LSTM.

2.1. Dataset

The ConvLSTM model was tested on two main datasets. They are station based IMD gridded data (Pai et al. 2014) and remote sensing based TRMM data (Huffman et al. 2016). Details of these datasets are given below:

The IMD dataset is obtained from interpolation of ground station data, over the Indian landmass, into a gridded form Pai et al. (2014). The dataset was created from the ground data obtained from various stations across India. The stations were chosen based on their density to avoid any inhomogeneities. The Shepard interpolation, based on weights calculated from distance to nearest grid point and direction, was applied for generating the interpolated values. This effort generated a ground-based daily gridded data with resolution of $0.25^o \times 0.25^o$ over Indian landmass, which was found to be more accurate than the other global gridded datasets (Rajeevan et al. 2006). We used this data, in this study, for the period 1974-2015.

NASA and Japanese Space Agency Jointly own the TRMM which contains the data obtained from satellite measurements and the same is available globally from 50^o N to 50^o S. The TRMM source data is in mm/hr unit, therefore a factor of 3 is multiplied to the sum for every grid cell. We have used the daily accumulated precipitation (mm/day) product for the period 1998-2015 from research quality 3-hour TRMM Multi-Satellite Precipitation Analysis (TMPA-3B42). The resolution of the data was $0.25^o \times 0.25^o$ having invalid values which were set as -9999. The TRMM accumulated precipitation is obtained as follows:

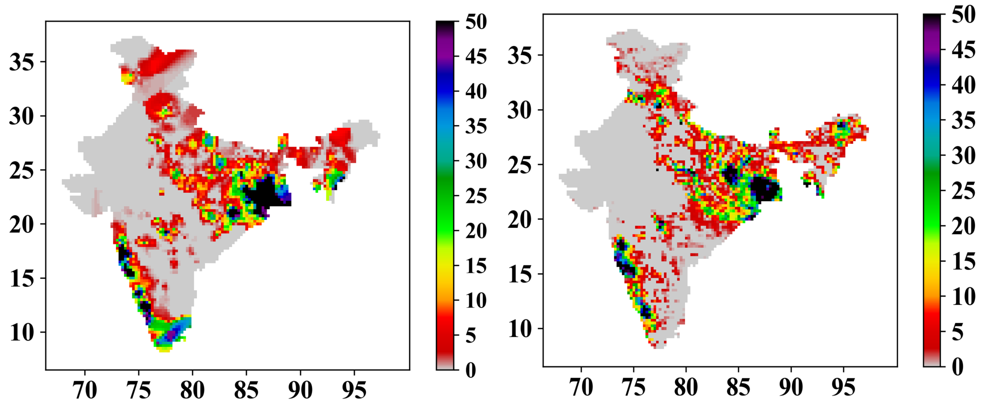


Figure 2. Sample total rainfall on a day (18-06-2011) from the IMD high-resolution dataset (in the left panel). The right panel shows a typical rainfall on a day from TRMM high-resolution dataset.

$$\Psi_d = 3 \times \sum_{i,j} (\psi_{i,j} \times \delta(\psi_{i,j})) \tag{1}$$

$$\Psi_{dc} = \sum_{i,j} = \delta(\psi_{i,j}) \tag{2}$$

Where $\psi_{i,j}$ represent the true value of the precipitation at a grid location (i, j). The $\delta(\psi_{i,j})$ =0 if the data point is absent otherwise 1. Ψ_d is daily precipitation value and Ψ_{dc} stands for the daily count of those values. The data set is available from January 1, 1998 to date. We have chosen the data for the present study till December 3, 2015 and between 6.375^o N to 38.625^o N and 66.375^o E to 100.125^o E. Thus, both data were utilized on a daily basis, with each frame reflecting the total rainfall of the day. A sample of total rainfall for a particular day from these datasets is shown in the Figure 2. For both datasets, separate models were developed (refer to sections 2.3 and 4).

2.2. Data processing

The ConvLSTM network used in this work receives data in 5 dimensions, namely: no. of samples, time steps, latitude, longitude, and variables. It is essential to clean up and prepare the data in a supervised learning format. During the processing of data for both datasets, different techniques described in the following subsections have been adopted.

2.2.1. Station based (IMD) Dataset

This dataset had several undefined values which were assigned as 'NaN'. There were some points which were assigned as 'NaN' in all frames and others were those which were rarely absent. The points under the second category were interpolated spatially from their closest neighbors. Special treatment had to be given to the points having 'NaN' in all frames to avoid losing spatial structure of the data while treating NaN values. We have implemented a new and efficient method for this problem, detailed in

Section 2.5.1.

2.2.2. Remote sensing based (TRMM) Dataset

There are a significant number of invalid points within the TRMM Dataset. We spatially interpolated them from the nearest neighbours. Although a high degree of skewness was a specific difficulty that was faced while dealing with this data, we wrote a custom loss function for TRMM data training. A similar approach was used in Shi et al. (2017). The details of the custom loss function are provided in subsection 2.4.1.

2.3. Models

We have mainly employed following three machine learning models

- ARIMA (Autoregressive Integrated Moving Average) is a forecasting model which works on the past values given in a time series.
- ConvGRU (Convolutional Gate Recurrent Unit) (Ballas et al. 2015) operates on the spatio-temporal data for forecasting purpose. It was developed for imagset.
- ConvLSTM method introduced by Shi et al. (2017) which is based on the LSTM method but operates on spatio-temporal data. Shi et al. used this method for precipitation nowcasting using radar data.

Each of these three methods was evaluated on the aforementioned data. The convLSTM algorithm was determined to be the most effective model based on the metrics discussed in section 2.4. Therefore, the ConvLSTM was chosen as the primary technique for this investigation. Architectural details of the ConvLSTM method is provided in the section 4.

2.4. Metrics for assessing the robustness of results

We used a new custom loss function (λ_{mse}) to deal with the invalid points in the TRMM dataset. One of the metrics used to validate our model is correlation coefficient (CC),calculated based on predicted and true values as given in equation 5. Furthermore, we calculated the ROC curves (using equation 6) to analyze the skill of the forecasts. The details of these metrics are provided in the following subsection.

2.4.1. A New custom loss function for TRMM data training

Since the TRMM data training is a regression problem, Mean Squared Error (MSE) is the usual choice of the objective function (loss function). However, the skewness in the data resulted in the model not predicting large values in the ground truth ($> 30mm$). It was quite important to capture the high rainfall values for our study. Therefore, the model was trained with a custom loss function (λ_{mse}) given as follows:

$$\lambda_{mse} = \frac{1}{N}\sum_{1}^{N}\sum_{1}^{L_1}\sum_{1}^{L_2} W_{n,i,j} * (\psi_{n,i,j} - \widehat{\psi}_{n,i,j})^2 \tag{3.1}$$

$$W = \ 1 \quad if \quad \psi_{i,j} >= 0.15 \tag{3.2}$$

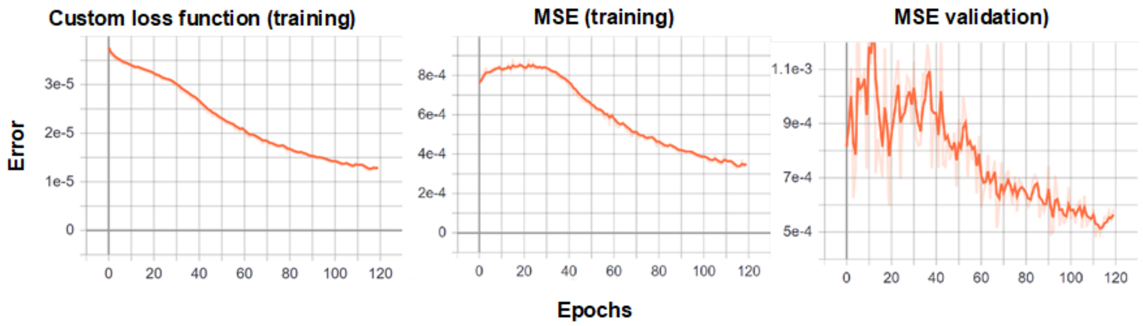


Figure 3. Comparing the variation of custom loss with MSE on TRMM Data (X-axis- epochs, Y-axis- Error).

$$W = 0.1 \quad if \quad \psi_{i,j} \ < \ 0.15 \tag{3.3}$$

Here ψ_{ij} represents the value from the TRMM data set which was normalized for model input using the min-max method. $\widehat{\psi_{i,j}}$ is predicted values of the precipitation. The L_1 and L_2 represent the total number latitude and longitude respectively. Total number of samples is denoted by N.

The choices for the hyper parameters in equations 3.2 and 3.3 were arrived at by trial and error approach. A higher weightage needed to be given to the higher value because the rainfall data was skewed and the extreme events were required to be captured appropriately. The choices of the limits and the weight are empirical. A comparison of the custom loss function with MSE defined in equation (4), is presented in Figure 3. In the figure X-axis and Y-axis represent epochs and error respectively. Training was stopped at early stage as shown because no further reduction in validation loss was found after those epochs. Use of the custom loss function allowed to not let the higher precipitation values ignored by the model. This type of approach has been adapted in the study by Shi et al. (2017) where they used a function called 'weighted loss function' to capture frequencies of different rainfall levels which were highly imbalanced.

$$MSE = \frac{\sum_N \left(\sum_{L_1} \sum_{L_2} (\widehat{\psi_{i,j}} - \psi_{i,j})^2 \right)}{N \times L_1 \times L_2} \tag{4}$$

It should be noted that the custom loss function (eq. 3.1) is used for TRMM data only. For the other data we have used MSE defined in equation 4.

2.4.2. Correlation, normalized RMSE and MAPE

In meteorology and geophysical fields, we generally get the data in the 3-dimensional space in particular, any variable in the data can be represented as $\psi_{i,j} \in [L_1, L_2, T]$, where T represents time. Thus, the data is in the form of space coordinates which represents spatial pattern maps in $[L_1, L_2]$ plane for a given time slice. One can have a temporal correlation between two variables at a given location, for a set of time coordinates, or alternately, for a given time, a correlation between the two variables for spatial locations. This metric is known as the pattern correlation coefficient (r_ψ). It signifies that for a given time how the spatial variances are related between two variables. In other words, it represents how well the two variables (say rainfall from observation and from forecast) are spatially collocated. It is calculated with the following formula (Weisstein 2020).

$$r_\psi = \frac{\sum\left[(\widehat{\psi_{i,j}} - \widehat{\mu})(\psi_{i,j} - \mu)\right]}{\sqrt{\sum(\widehat{\psi_{i,j}} - \widehat{\mu})^2 \ \sum(\psi_{i,j} - \mu)^2}} \tag{5}$$

Here, μ represents the mean value of original data and $\widehat{\mu}$ is mean value of the predicted data. The summation is taken over the test data. We calculated this metric for TRMM data set, with corresponding IMD data, shown in the results section. Apart from the custom loss function for the TRMM data, we have used an efficient approach for dealing with 'NaN' values in IMD data, described in the next section. We also calculated the normalized root mean squared error (NRMSE) at each grid by dividing the RMSE with the standard deviation for the entire test and training data. Similarly, the MAPE has been calculated for each grid.

2.4.3. Receiver Operating Characteristics (ROC) curve

Another metric, used in this work, to validate the results is ROC. It is an important tool for forecast verification and decision-making processes. It is a plot which illustrates the diagnostic ability of an forecast classification system, using its varying discrimination threshold (see (Marzban 2004)). The ROC curve is created by plotting the hit rate or True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold settings. The ROC analysis provides the ways to select possibly optimal models and to discard suboptimal ones independently from (and prior to specifying) the cost context or the class distribution. This analysis is related, in a direct and natural way, to cost-benefit analysis of diagnostic decision making. Hence, it is a standard method of forecast skill analysis for operational rainfall forecast. While the correlation method can't discriminate the threshold criteria for more false positive occurrence, the ROC method can do so. The ROC method applied here shows a better fidelity of the proposed model. The formula for calculating these rates are given in equation (7).

$$TPR = \frac{TP}{N_H} \tag{6.1}$$

$$FPR = \frac{FP}{N_L} \tag{6.2}$$

Where, TP denotes True Positive and it is number of days when both area averaged values of ground truth and prediction are above average. N_H is the number of days when area averaged ground truth values is higher than the threshold values chosen based on minimum and maximum rainfall values in the data. FP denotes False positive and it represents days for area averaged value of prediction above the threshold when the prediction value is below the level. The number of days when area averaged ground truth rainfall values lower than the threshold is represented by N_L in the equation 6.2.

2.5. Data preparation

For IMD data, we have chosen 22 years of data for training and 5 years for testing. For the TRMM dataset, 12 years were set for training while 5 for the model testing (refer table 1). Both data sets have 1 day time resolution. With a window size of 5 days, we used moving windows method for input data generation, comparable to studies using RNN architecture (Lara-Benítez et al. 2021). Depending on the situation, the window moves throughout the training or test period (train and test durations are specified in the table 1). With an input sample of 5 days of spatial data (captured by the moving window) and a corresponding label of spatial rainfall data for the day after the window, the data is converted to a supervised learning format for training. Each sample in the input has a dimension (timesteps, channels, rows, columns) and a 2-dimensional (rows, columns) format for the corresponding label.There are 5 timesteps in each input sample, 1 channel (only the precipitation variable is used in the input), and rows × column(lat × lon) =129*135. Both the IMD and TRMM datasets have the same input dimension.

Table 1. Details of the data segregation for training and testing purposes.

Data set	Training set	Testing set
IMD	1975-1996	2011-2015
TRMM	1998-2009	2011-2015

2.5.1. Method For Dealing With undefined ('NaN') Value

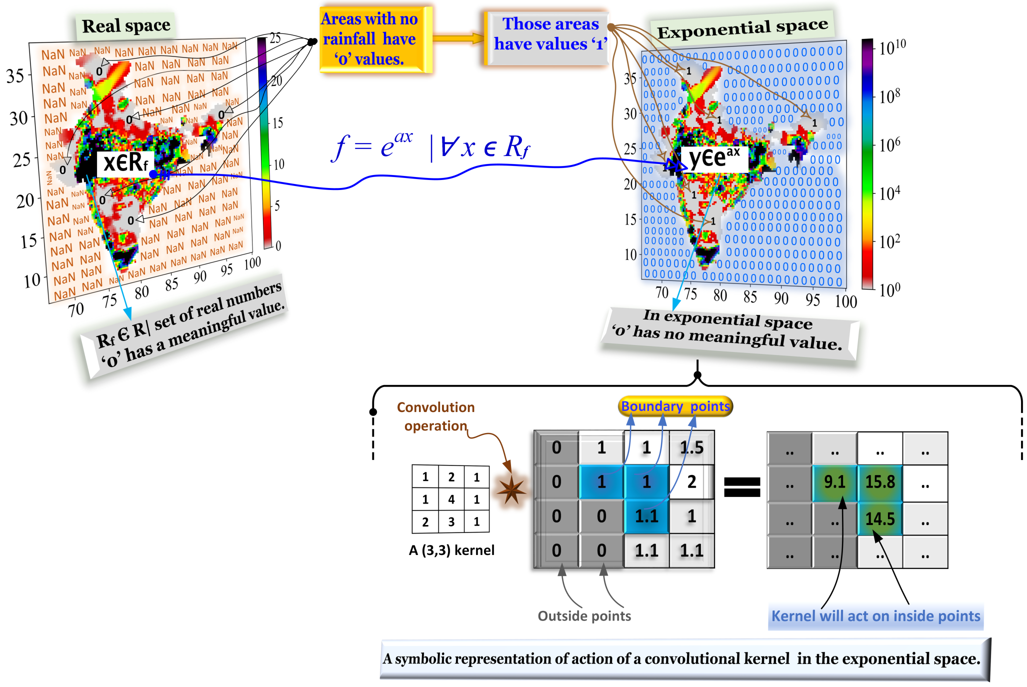
The given IMD dataset has undefined precipitation values marked as 'NaN'. To deal with these 'NaN' values, a new strategy was employed in this work. A detailed description of the strategy is provided in this subsection.

As mentioned in section 2.2.1, two kinds of 'NaN' values were present in the data: (i) points which are 'NaN' in all frames which refers to those points which correspond to ocean and lie outside India and, (ii) points that are occasionally missing due to lack of observation on a day because of equipment malfunctions etc. The occasionally missing 'NaN' points were spatially interpolated from their nearest neighbours values.

The 'NaN' values (in point (i) above) cannot be extrapolated as there is no sufficient data for so many points. Also they can't be replaced by '0' because '0' number has a significant value for precipitation as it indicates no rains (depicted in Figure 4). Therefore, in the real space, data would furnish wrong information to the model. In this study, a new method was tried out to deal with 'NaN' values falling outside of the Indian landmass. This involves taking the data into exponential space and then assigning '0' for missing values represented by 'NaN'. This is done keeping in mind the practice that; in general, it is safe to input missing points with '0' provided that it doesn't represent a meaningful value. The condition of '0' not being a meaningful value is not met in the real space because the locations with no rainfall are marked as '0' in the raw data. Hence, an efficient transformation was required which we chose as exponential space as discussed in the previous paragraph and illustrated in Figure 4.

While converting the whole data from 'real space' to 'exponential space', we got rid of the issue of wrong information (Figure 4). The network learns from exposure to the data to treat the value '0' as missing and start ignoring them in the transformed space (Chollet 2017). We found that this method is best suitable for the model training in the present scenario since it is one of the effective techniques which can take care of sharp gradients in spatial patterns of rainfall that we see along the west coast of India. In meteorological data, dealing with the missing values is an essential problem and the said transformation is one of the effective methods which can be used operationally. With this transformation the data preparation is done, as explained below (also refer to Figure 4).

1. Identify all non-NaN points and normalize them with maximum value in the


Real space
Areas with no rainfall have 'o' values.
Those areas have values '1'
Exponential space
$f = e^{ax} \mid \forall x \in R_f$
$x \in R_f$
$y \in e^{ax}$
$R_f \in R$| set of real numbers 'o' has a meaningful value.
In exponential space 'o' has no meaningful value.
Convolution operation
Boundary points
A (3,3) kernel
Outside points
Kernel will act on inside points
A symbolic representation of action of a convolutional kernel in the exponential space.

1	2	1
1	4	1
2	3	1

0	1	1	1.5
0	1	1	2
0	0	1.1	1
0	0	1.1	1.1

..	..	..	..
..	9.1	15.8	..
..	..	14.5	..
..	..	..	..

Figure 4. Illustration of data transformation from real space to exponential space. The conversion in exponential space has the benefit of considering '0' during model training as it doesn't possess any meaningful value.

dataset (for all points).

2. Apply the transformation $f = e^{ax} \forall x \in \Re |$ $\Re$ is set of real numbers having no NaN.
3. Substitute zero for all NaN locations in the rearranged dataset (legitimate values now range from 1 to e^a).
4. The choice of 'a' is appropriate when the range of the initial dataset and transformed dataset approximately match. In this way '0' rainfall value is transformed to 1 so the whole range of allowed values becomes $[1, \infty)$.
5. Network maps input to output in exponential space.
6. The spatial structure of data is preserved; hence spatial correlations can be learned.

In our knowledge, no models in literature describe treating such missing values (i.e., where observation values are unavailable) in an effective way and it is the first time such a transform has been used in the field of meteorology for AI model training. Hence, this method may be treated as a novel approach to deal with missing data values.

The overview of the data and methodology adopted in this study and describes in the current section is depicted in the figure 5.

3. Problem Formulation

Usually, weather predictions come with probabilistic scoring, which is why problem statements of weather prediction can be written as most likely N-sequence selection from an ensemble of prediction. As a spatiotemporal sequence forecasting problem (for

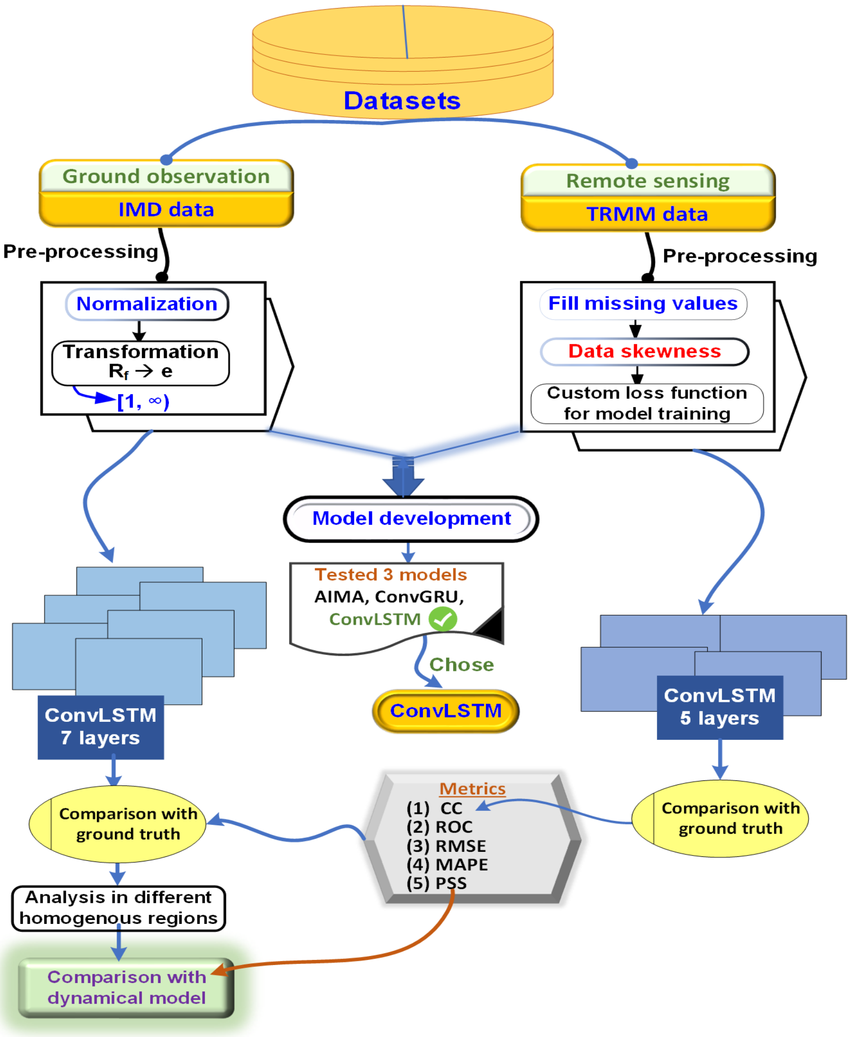


Figure 5. An overview of methodology used in this study.

monsoon rainfall), our input state can be represented as vectors of variables over a spatial grid of $L_1 \times L_2$ locations as described in Section 2.4.2.

On these locations, say, total N_p variables are measured. Therefore, any observation at a given time is represented in a mathematical space $R^{(L_1 \times L_2 \times N_p)}$, where R is the domain of the observed variables. Given a certain amount of the past data, it can be represented as a sequence of elements from this aforementioned space as $\Psi_1, \Psi_2, \Psi_3, \cdots \Psi_t$, where $\Psi_n = \psi_{i,j}$ is precipitation value at a particular grid location. Then the forecasting problem is defined as to predict the least error K-length sequence in the future given the previous 't' observations (including the current one) as input. Following Shi et al. (2015), this can be represented as

$$\widehat{\Psi}_{t+1}, \cdots, \widehat{\Psi}_{t+k} = \underset{\Psi_{t+1},\cdots\Psi_{t+k}}{\mathbf{arg\,max}}\; p\left(\Psi_{t+1}, \cdots, \Psi_{t+k} | \Psi_1, \Psi_2, \Psi_3, \cdots, \Psi_t\right) \tag{7}$$

where $\widehat{\Psi}$ is the predicted output sequence. In other words, our problem reduces to finding a suitable architecture among various possibilities of hyper-parameters and layer choices which reduces the error between the predicted and the ground truth of observations. We started with simple ConvLSTM-based architecture and tuned it to improve our predictions but were constrained by the number of layers and layer-specific hyper-parameters that could be chosen given an upper limit of RAM and processing power of the Graphics Processing Unit (GPU).

4. Details of ConvLSTM architecture used in this study

As mentioned before, we decided to employ the ConvLSTM method, thus developed the model for this algorithm and carried out several experiments to mature the architecture. The experiments were mainly based on data pre-processing and techniques used for handling the undefined rainfall values assigned as 'NaN'. In the case of IMD data, we used the exponential space to train the model. Once the algorithm and kind of Neural Network architectures are decided, the network's fine-tuning, called hyper-parameter optimization, is accomplished. Various combinations of kernel sizes, number of filters, activations, number of layers, optimization algorithm, and learning rate are tried out during training before asserting the best final architecture. For both datasets, the developed models were trained using the Keras API with TensorFlow running as a backend. The choice of the last layer to be fitted to the ConvLSTM output was selected from the following options:

1. Conv3D Layer: This layer is applied to the 5-dimensional sequential output of the connected ConvLSTM layers. It performs a 3D convolution over space and time dimensions to produce the final output.
2. Conv2D Layer: To apply this layer, the ConvLSTM is set to return only the output corresponding to the last time-step in an input. Therefore, this layer uses a 2D spatial convolution on the spatial dimensions alone to give the output.
3. Locally Connected 2D Layer: This layer acts similar to Conv2D but in a generalized form. The kernel used is different at each location throughout an image. It has more parameters compared to Conv2D, but spatially localized patterns could be learned.

The developed model used the Conv2D as the last layer based on the MSE value. A comparison of MSE among different layers is provided in Table 2.

Table 2. A comparison of MSE among different layers used as final layer. The least value was obtained using Conv2D layer, hence, it was chosen as last layer.

Layer	Conv3D	Locally connected 2D	Conv2D
MSE	3.1×10^{-2}	2.94×10^{-2}	2.76×10^{-2}

This study is an attempt to provide a proof of concept for applying the ConvLSTM method for ISMR forecasting. The study by **?** proved that this method is better than other state of art machine learning methods available for forecasting meteorological variables. The details of the model architecture used for IMD and TRMM data are summarised in Table 2 and Table 3. The total number of parameters trained for IMD and TRMM datasets are 43559 and 284409, respectively.

Table 3. The model architecture used for training on the IMD rainfall dataset. Total 7 layers were used for this model.

LN	Layer name	Architecture type	Activation	Kernel size	# Filter
1	ConvLSTM2_1	Convolutional LSTM	tanh	(3,3)	4
2	ConvLSTM2_2	Convolutional LSTM	tanh	(3,3)	8
3	ConvLSTM2_3	Convolutional LSTM	tanh	(3,3)	8
4	ConvLSTM2_4	Convolutional LSTM	tanh	(3,3)	16
5	ConvLSTM2_5	Convolutional LSTM	tanh	(3,3)	16
6	Conv2D_1	Convolutional	relu	(3,3)	15
7	Conv2D_2	Convolutional	relu	(3,3)	1

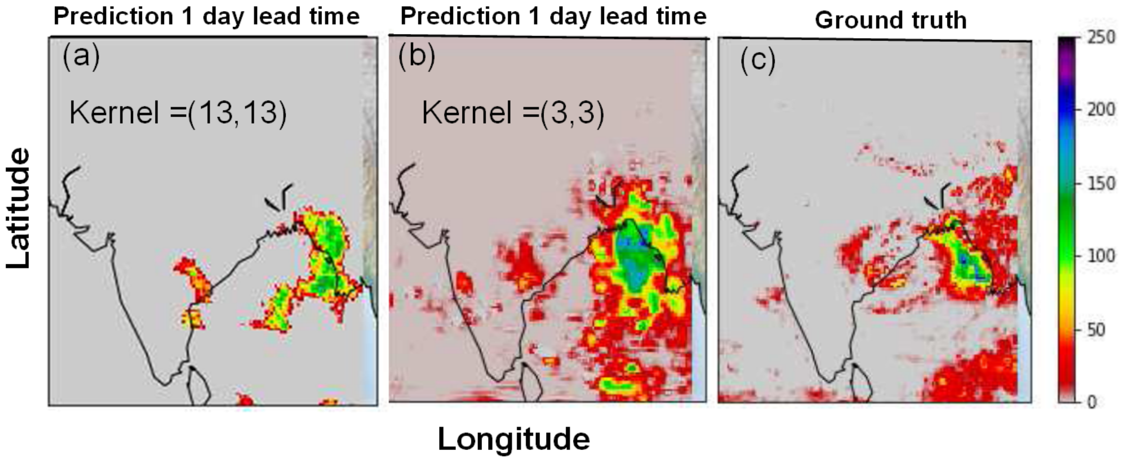


Figure 6. Comparing the 1 day lead predictions from a model using kernel sizes (13,13) and (3,3) with Ground truth, respectively (on TRMM Data).

4.1. Kernel size optimization

Furthermore, we did experiments with different kernel sizes. It was observed that smaller kernel sizes tend to do better than larger ones. An example for 1 day lead time prediction is shown in Figure 6 when the Kernel is large (13, 13) and one with small (3, 3). Figure 6 suggests that a smaller kernel of size (3, 3) can capture larger values effectively and also over more regions as compared to the larger one (13, 13).

4.2. Hyper-parameters

The hyper-parameters used in both data sets are provided in the table 5. The learning rate and number of epochs are different for both data sets. The Adam optimizer was used for the adaptive estimation of first-order and second-order moments.

Table 4. The model architecture used for training on the TRMM data set. Total 5 layers were used for this model.

LN	Layer name	Architecture type	Activation	Kernel size	# Filter
1	ConvLSTM2_1	Convolutional LSTM	tanh	(3,3)	8
2	ConvLSTM2_2	Convolutional LSTM	tanh	(3,3)	12
3	ConvLSTM2_3	Convolutional LSTM	tanh	(3,3)	6
4	Conv2D_1	Convolutional	relu	(3,3)	6
5	Conv2D_2	Convolutional	relu	(3,3)	1

Table 5. The hyper-parameters used for IMD and TRMM data sets with (lat, lon)= (129,135).

Epochs	500 (TRMM = 200)
Learning rate	10^{-4} (TRMM = 10^{-3})
Optimizer	Adam ($\beta 1 = 0.9, \beta 2 = 0.999$)
Stride	(1,1) for each layer
Dropout rate	0
Timesteps in each sample	5
Tensorflow	2.2.0
Keras	2.4.3

5. Results and discussion

We solved a regression problem rather than classification as described in equation (7) in the section 3. However, classifications are made to understand the fidelity of the generated forecast. Normally, it is known that forecasts are skillful for rainfall above or below certain amplitude (or certain frequency). Verification of meteorological forecast is made in multi-category classification to emphasize the more skilful category. Operational forecasters always require such information to see the reliability of the forecast when the output values are above a certain threshold. The categories are made based on standard World Meteorological Organization manuals. As mentioned in the section 2.3, we considered five different metrics to verify our forecast. The results obtained from model and analysis of metrics are presented in this section.

5.1. Model comparison

As discussed in the section 2.3, we have employed three different machine learning algorithms on these data sets. One of them is the baseline method ARIMA and rest of two are ConvGRU and CnvLSTM. The ConvGRU is a new generation of RNN ans is more straight froward then LSTM. We evaluated the performances of these models by comparing the metrics. In this, comparison, the ConvLSTM method stood as best method to forecast the rainfall values. Thus, we employed the convLSTM method on two sets of data: IMD gridded data and remote sensing TRMM data. Since both data sets have different pre-processing, as discussed in section II, two separate models were developed for them and were tested on validation data as given in Table 1. We compared the correlation coefficient (r_ψ) for these three methods as depicted in figure 7. The ConvLSTM method has better correlation and similar NRMSE among these three methods. Thus, the ConvLSTM has been found to be the winner. As a result, in this study, we chose the ConvLSTM method as main method for precipitation forecasting.

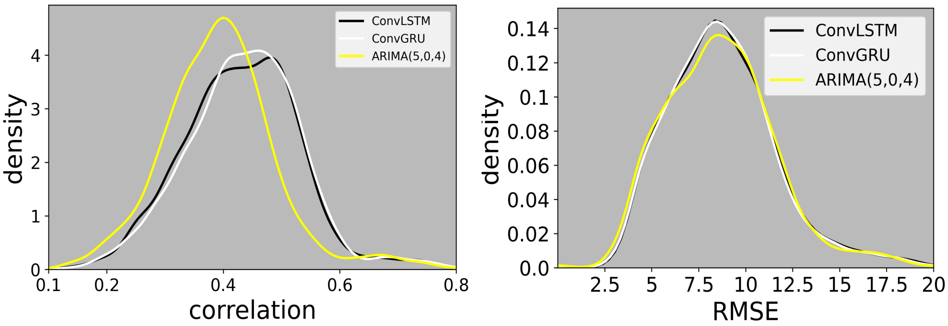


Figure 7. Correlations obtained using the ConvLSTM, ConvGRU, and ARIMA methods are compared (left panel). The right panel displays the RMSE pdfs for all three methods, which are nearly identical.

5.2. Comparison with Ground Truth

The outputs obtained by applying model on both data sets were compared with available ground truth.

5.2.1. IMD homogeneous regions:

First, we analyzed predicted data from the model for the homogeneous regions defined by the Indian Meteorology Department (IMD) (Kothawale and Rajeevan 2017). There are a total of 6 homogeneous rainfall regions categorized based on the rainfall percentage in monsoon seasons during the period from 1871-2016. We calculated the Coefficient of Correlation (CC) for area-averaged data for 5 years' time series and area-averaged rainfall for the 5 years duration from 2011-2015 for the IMD homogeneous regions. A comparison of these metrics with ground truth and model data for the west-central region is shown in figure 8.

Table 6. List of skills metric for different homogeneous regions. The correlation coefficient (CC) drops from 0.79 (West Central) to 0.52 (South Peninsular). There is no specific trend for RMSE values.

Region	CC (RMSE) 1^{st} day	CC (RMSE) 2^{nd} day
West Central	0.79 (3.70)	0.56 (5.18)
Central NE	0.7 (3.92)	0.42 (5.11)
Northwest	0.76 (3.77)	0.58 (4.64)
Hilly Regions	0.53 (4.19)	0.24 (4.93)
Northeast	0.55 (5.85)	0.3 (6.84)
South Peninsular	0.52 (4.15)	0.31 (4.46)

It is to be noted that the model can capture the rainfall up to 2 days lead time in the central region. A similar comparison for the Central North East region is provided in figure 9. The skills for other homogeneous regions are presented in the Table 5. The CC values in this table varies between 0.79 over West Central region to 0.52 over South peninsular. The CC and Root Mean Square Error (RMSE) values, obtained from this

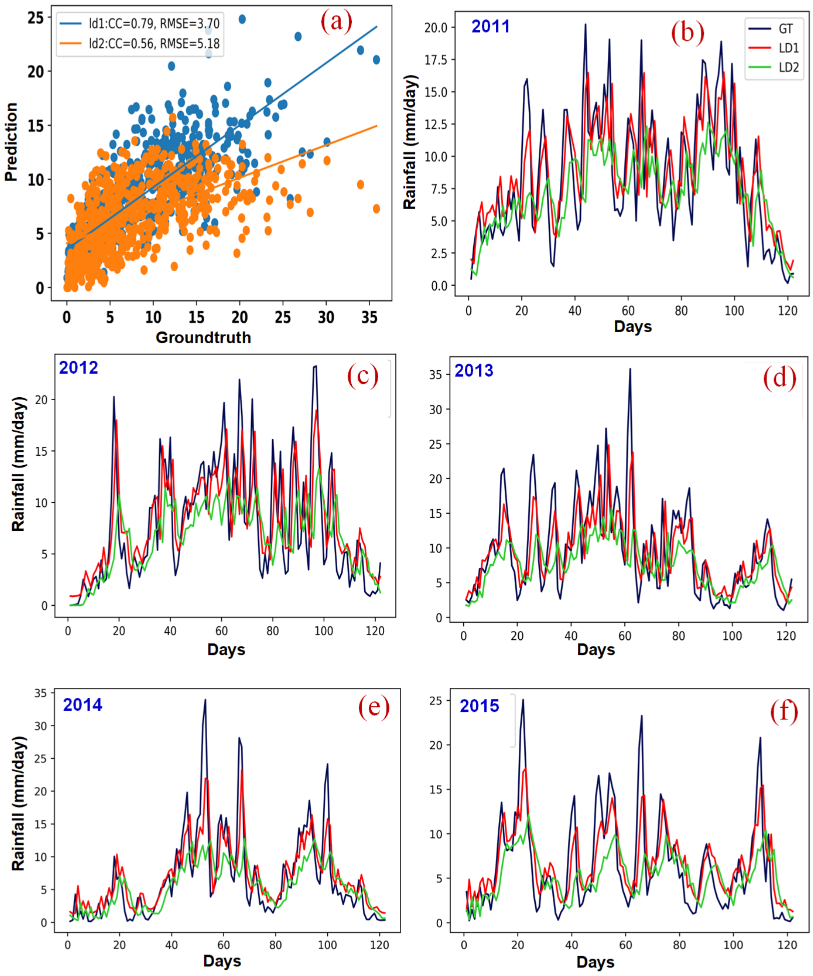


Figure 8. : Comparison of area-averaged correlation coefficient (CC) and RMSE (panel a) for 5 years time series (IMD) data and area-averaged rainfall for 5 years duration (2011-2015) for West Central region (panel b-f). The days on X- axis starts from 1^{th} June and ends at 30^{th} September (the JJAS period). The Ld1 refers to lead day one and similarly Ld2.

model, are comparable to state-of-the-art dynamical models such as present as shown by (Mukhopadhyay et al. 2019).

5.2.2. Comparison using entire landmass area

Calculating the area average rainfall and comparing it with the ground truth for the homogeneous region is one way to test the model's accuracy. Further, we compared the spatial pattern of the forecast skill of the precipitation forecast for up to 2 days lead time for IMD and TRMM data for every grid point. One such comparison is depicted in Figure 10. The TRMM dataset can capture localized as well as large-scale organized precipitation patterns. Previous studies have noted the capability of TRMM derived precipitation in capturing rainy spells and the extremes. It is beneficial over the regions of complex topography where in-situ data are often not available. However, it also predicts some false positives, predicting rainfall at places, not in the ground truth. The data was taken for August 8, 2011, for IMD, and August 7, 2011, for TRMM. The difference of 1 day between TRMM and IMD is due to the convention

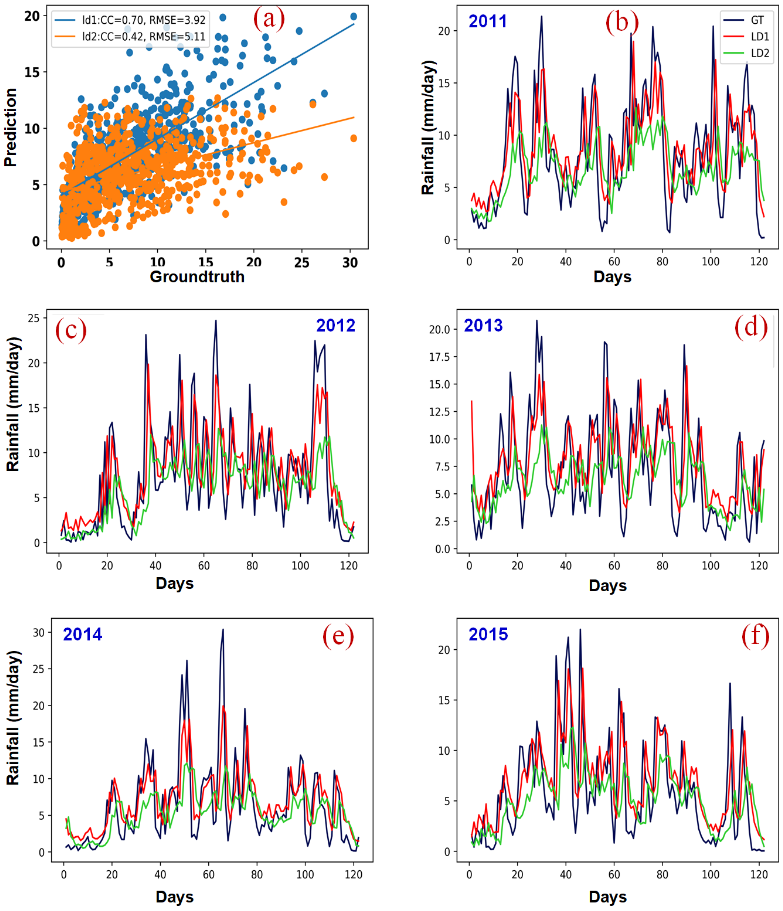


Figure 9. : Comparison of area-averaged correlation coefficient (CC) and RMSE (panel a) for 5 years time series (IMD) data and area-averaged rainfall for 5 years duration (2011-2015) for Central North East region (panel b-f). The days on X- axis starts from 1 June. The time period is JJAS.

that IMD rainfall for a day is the rainfall obtained in the last 24 hours of the recorded time, while for TRMM, it is the rainfall in the next 24 hours of the recorded time.

The ISM rainfall shows significant variability in space and time. On some occasions when the monsoon is in 'active or organized' phase, the rainfall patterns are widespread in space while during the 'break or weak' phase we see isolated spells across the region (Singh et al. 2021a) . It is to be noted that the rainfall memory (in time) is less as compared to other meteorological variables (e.g. temperature). Our aim here is to understand how well the model retains this memory and produces rainfall in space and time. Figure 10 compares the 1 and 2 day lead predictions generated by the model with the IMD and the TRMM data. It is to be noted that the training of the model was performed for both sets of data (the training periods were different). Therefore while comparing the model forecasts, corresponding observations are also used. The observation days here correspond to the model lead days and the bias is nothing but the difference (in space) between the observed rainfall for that day and the corresponding model forecast. It is seen that overall biases in both first (denoted Ld1) and second day (denoted as Ld2) lead times are smaller for the IMD data compared to the TRMM data. Though rainfall over the core monsoon zone shows less bias, there is significant

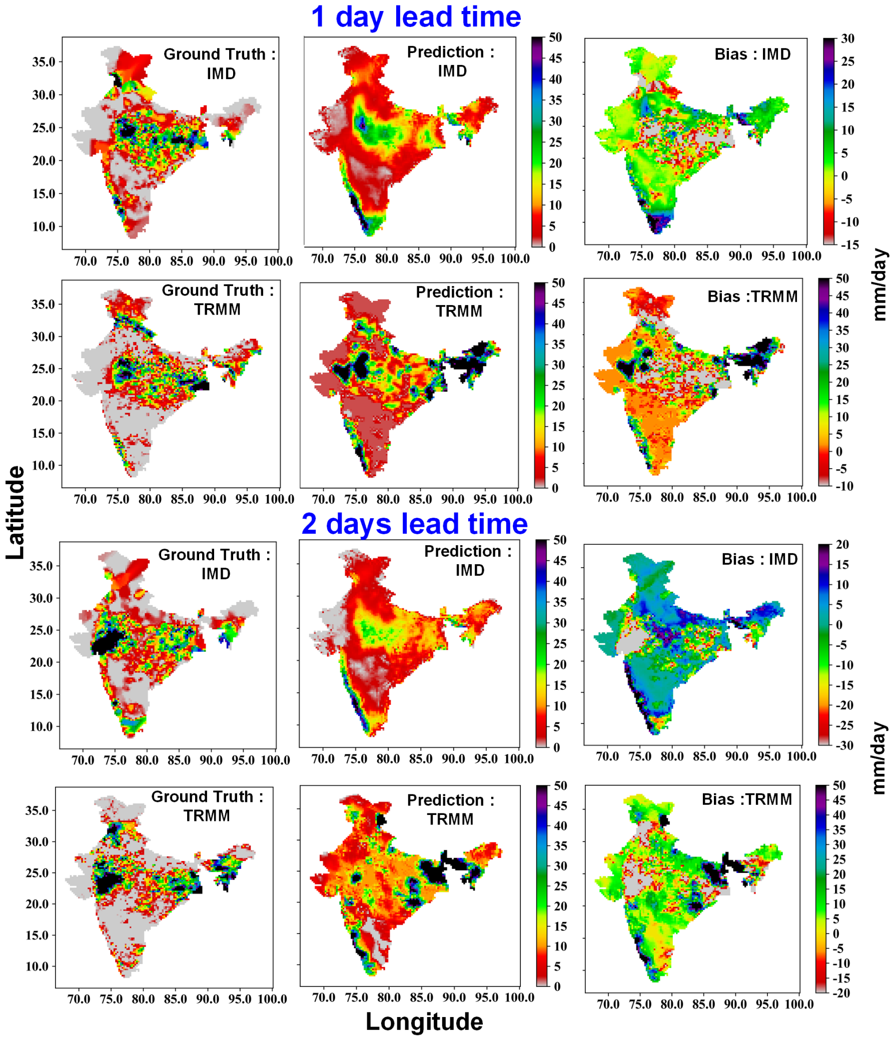


Figure 10. : Comparing the 2 days lead predictions from the ground truth. The upper panels to show the comparison for IMD data, and the lower ones are for TRMM data in both Ld1 and Ld2 cases. The plot for Ld1 is for August 8, 2011 (IMD), and August 7, 2011, for TRMM. The plots in the last columns represent biases. A similar comparison was present output obtained from dynamical model in Huffman et al. (2016).

bias over the regions of high elevations for both cases (e.g. over the Western Ghats, the Himalayan region). The analysis presented here helps us to identify the regions where the model has good or poor fidelity in reproducing the actual rainfall and also indicates the spatial coherency between the two.

5.3. Comparison of CC, NRMSE and MAPE

We calculated the pattern correlation coefficient (CC), NRMSE and MAPE as described in section 2.4.2 for both datasets.

5.3.1. IMD Data

The pattern correlation and normalized RMSE (NMRSE) obtained from the IMD data is shown in Figure 11. It is seen that pattern correlation worsens from lead day 1

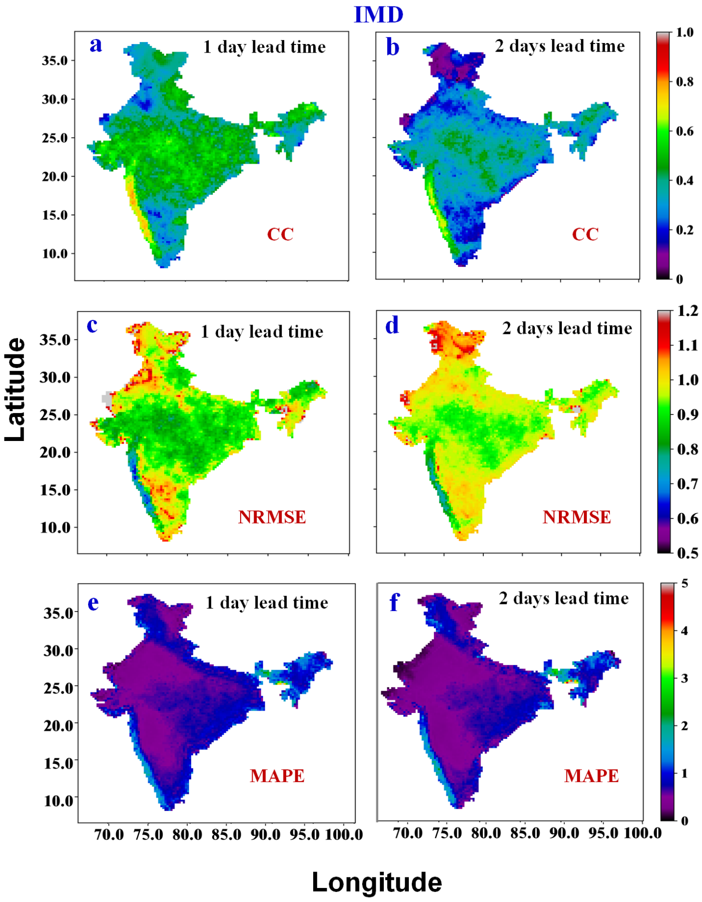


Figure 11. Correlation (panels a & b) , NRMSE (panels c & d) and the MAPE (panels e & f) of Ld1 and Ld2 for IMD data.

to 2. Further, the pattern correlation shows large variations across the Indian region. The best correlations are noted over the west coast and monsoon trough region, while the lowest values are noted over the northern regions. The 2 lead days' patterns are reasonably correlated over the Western Ghats and monsoon trough region (0.8 on lead day 1 and 0.6 for lead day 2). However, the model fares poorly over the parts of Himalayas regions and Rajasthan.

Over these regions, the pattern correlations deteriorate quickly after lead day one (Ld1) and reach below 0.4 on lead day 2 (Ld2). One plausible reason behind the poor correlation over these regions might be the sparse density of IMD stations (as mentioned in Pai et al. (2014)).

Nevertheless, the model reasonably captures the variability in the short term. The normalized RMSE for the Ld1 and Ld2 are shown in the lower panel of Figure 11. The normalized RMSE (NRMSE) is considerably low in most regions except in some parts of the North East area (around the Sikkim region). Relatively higher values of NRMSE can also be seen in the Western Ghats area for both Ld1 and Ld2. The model's performance is comparable to state of art numerical weather prediction models (Rao et al. 2019; Rajeevan and Santos 2020). Similarly, the mean absolute percentage error (MAPE) is about 1% in the entire except the Sikkim region.

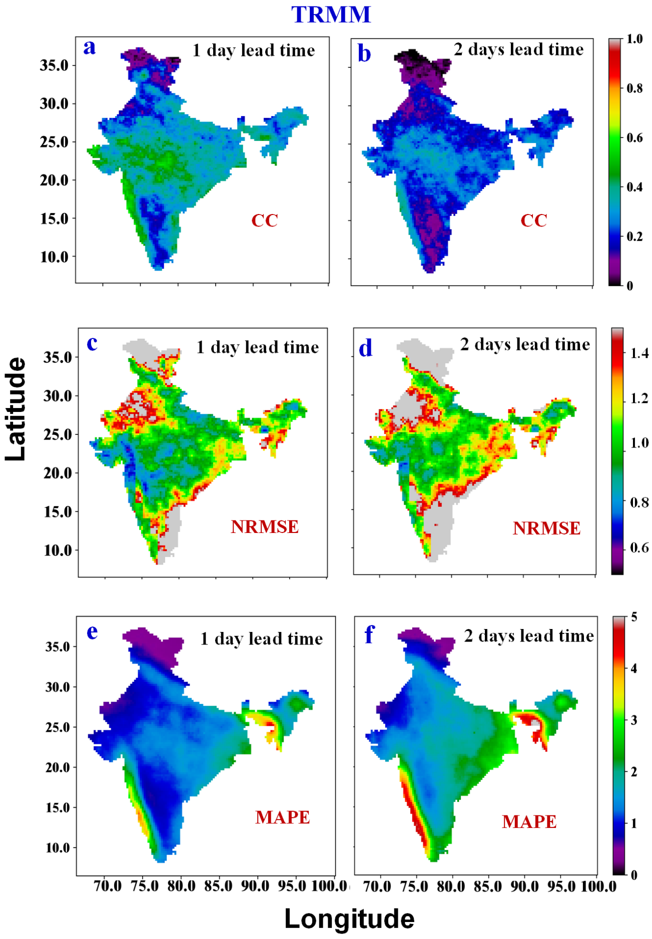


Figure 12. Correlation (panels a & b), normalized RMSE (panels c & d) and the MAPE (panels e & f) of Ld1 and Ld2 for TRMM data.

5.3.2. TRMM data

In TRMM data case, with increasing lead time, the pattern correlation decreases significantly as shown in Figure 12 (panels a & b). The model requires improvements to capture the rainfall for TRMM data better. One possible improvement can be to use multivariable input for training. The NRMSE for Ld1 and Ld2 for TRMM data are presented in lower panels (panels c and d) of Figure 12. It is noted that the Western Ghat area has higher CC and lower NRMSE. The MAPE, presented in the panels e & f have higher values in North-east regions for both Ld1 and Ld2.

5.3.3. Homogeneous regions of IMD data

The pattern correlations for homogeneous regions show a similar trend as in the entire Indian territory, which means it deteriorates after day 1. Figure 13 depicts the pattern correlations for West Central (panel a) and Central North-East (panel b) regions. A better CC was obtained in the Central NE area for the Ld1.For the second day lead time, the CC falls quickly in both areas as shown in Figure 13. Further, we calculated the PDF of RMSE for Ld1 and Ld2 of heavy rainfall in these regions. The heavy rainfall days were selected by taking only those days in which atleast 10 percent of

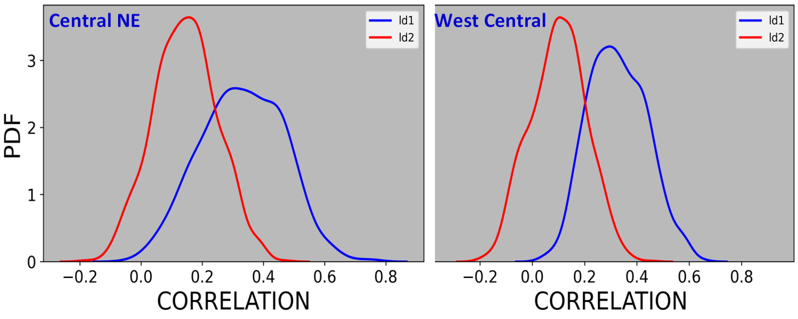


Figure 13. Comparison of the pattern correlation for Central North East (left panel) and West Central (right panel) for 2 days lead time for the IMD data.

the grid points in the homogeneous region had more than 95 percentile rainfall value. A comparison of PDFs is provided in Figure 14. The Ld1 RMSE is found to be less than Ld2 for three homogeneous regions, namely, Central NE, West Central and North East. There was no difference in RMSE between Ld1 and Ld2 forecasts was found for the other three regions.

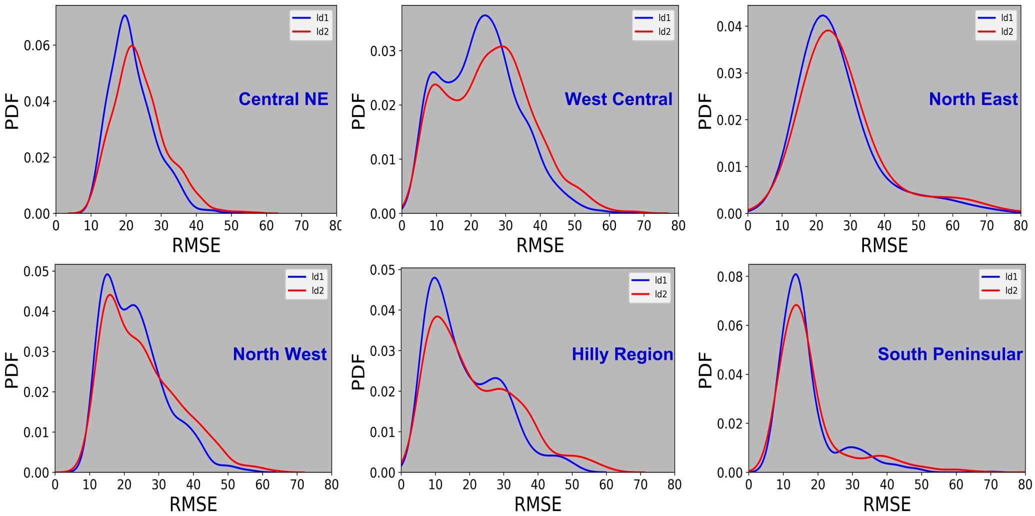


Figure 14. A comparison of PDF of RMSE calculated for lead day 1 (Ld1) and lead day 2 (Ld2). The RMSE for Ld1 is found to be less than Ld2 in the upper panels representing three regions: Central NE, West Central and North East.

5.4. Comparison of Receiver Operating Characteristics (ROC) curve

Another skill metric calculated for homogeneous regions is receiver operating characteristics (ROC), defined in section 2.4. A description of the application of the same method is provided in Marzban (2004), highlighting it as a measure of classification performance.

In our study, we have used a simple skill verification method as well as category (or threshold) based classifier verification. We calculated TPR and FPR (equation 7.1) for rainfall values in all six regions after binning the rainfall in different categories. The categories are determined based on minimum and maximum rainfall values and then slices them in 1mm intervals. Category-wise comparison indicates the skill of different rainfall bins, thus giving an idea on how the skill varies in different rainfall categories. Comparisons of these rates for all regions are provided in Figure 15. The blue dots indicate Ld1 forecast and orange dot represent the Ld2 forecast. The blue curve has larger Area Under the Curve (AUC) values, consistent with the correlation values (i.e. skill of Ld1 greater than the skill of Ld2) for these regions. The North West region does not show much difference in Ld1 and Ld2 skill.

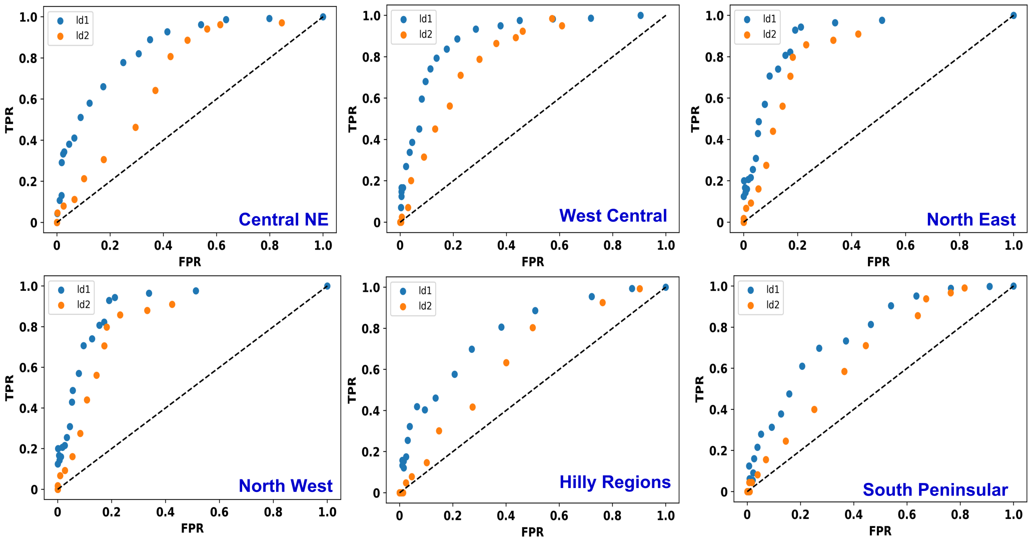


Figure 15. Comparison of ROC skill for different homogeneous regions. The Ld1 TPR is better for three regions: Central NE, West Central and North West.

5.5. Comparative Skills vs state-of-the-art operational numerical model

We have also compared the skill score of the employed model to that reported for a state-of-the-art numerical model Rao et al. (2019) for GFS T1534. For this purpose we have compared the Peirce Skill score (PSS) (Manzato 2007) as obtained from the ConvLSTM model and to that of a sophisticated numerical model (see Fig 3(b) of Rao et al. (2019)). Figure 16 illustrates one such comparison for the year 2011. For Ld1, the PSS skill obtained from the ConvLSTM method is better to GFS up to a 15 mm rainfall threshold, whereas for Ld3, the PSS skill derived from the ConvLSTM method is stronger up to a 6 mm rainfall threshold value. A drop in skill score for various rainfall thresholds is depicted in figure 3(b) of Rao et al. (2019). We observed that the skills for Ld1 and Ld2 based on ConvLSTM method, for the year 2011, have superior or comparable skill for at least wet and moderate spells. We have also compared the PSS for years 2012-2015 and obtained results (see the supplementary figure S1) with good skill.

5.6. Using multi-variables

Multivariate learning is essential to capture the low-frequency variability of rainfall as low-frequency sub-seasonal waves are convectively coupled waves with moisture, the surface low-pressure, and wind. We also tested the model for some more variables and

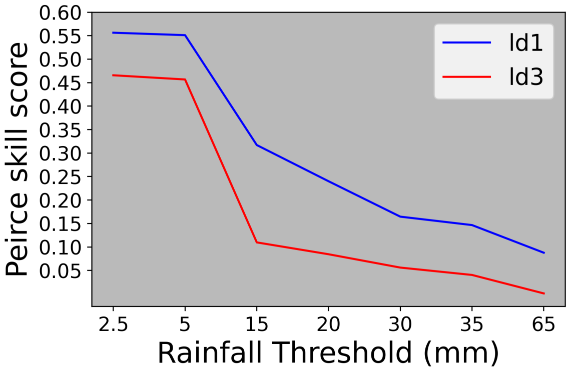


Figure 16. Peirce skill Score (PSS) obtained from ConvLSTM method for lead day 1 and lead day 3 forecast. The PSS can be compared with the same obtained from GFS T1534 cf. Fig 3(b) of Rao et al. (2019).

found little improvement with 6 variable inputs as depicted in figure 17. These six input variables are (i) rainfall, (ii) orography, (iii) speific humidity at 700 hPa(q700) (iv) at 850 hPa (q850) and (v) specific pressure and soil moisture.

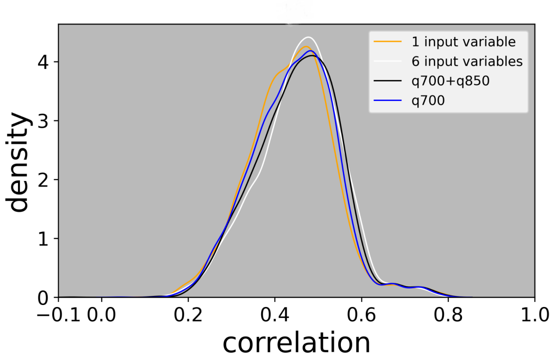


Figure 17. Comparison Ld1 correlation for multiple variables.

The figure indicates that the majority of the improvement in the six-variable input model can be accounted for by just two variables (q700 and q850).

6. Conclusions

This study focused on implementing a deep learning model, for the short-range forecasting of the ISMR. Three deep learning models, ARIMA, ConvGRU, and ConvLSTM, were tested on two separate datasets constructed using ground observation (IMD) and remote sensing techniques (TRMM). The ConvLSTM model was found to be the best method among three. ConvLSTM based models have been used for short-range forecasts/nowcasts in literature with some success. The proposed model is a proof-of-concept which can capture the spatio-temporal structure of the forecast data.

The convolution operation is not well-defined in the literature when we do not have data over a certain spatial domain. The IMD data, for example, do not have values over the ocean. Such sharp gradient at land-sea boundaries can be potentially problematic for convolution operation due to the absence of data. We applied an efficient approach and tackled the undefined values (i.e., grids having no data). For which the data were first transformed from real space to exponential space. The model training was done in exponential space allowing rainfall values to span $(1, \infty)$; replacing the NaN values with '0', as '0' was no more significant value in this space. The remote sensing TRMM data has skewness and model was not able to capture the high rainfall values. To resolve this issue, a custom loss function has been defined.

The model-produced forecast shows reliable skill with observations (the ground truth); however, up to 2 days lead time only. The model performance was evaluated using five metrics: (i) CC, (ii) NRMSE/RMSE, (iii) MAPE, (iv) ROC and (v) Peircs Skill Score (PSS). The efficiency quickly goes down after that, as seen in the pattern correlations. A low correlation is seen at the northern and North Western parts along the east coast of India. The forecast is also done separately for homogenous monsoon regions described in the Kothawale and Rajeevan (2017). In this case, the area-averaged correlation for 5 years' time series is found to be reasonably good, and the RMSE for this data is significantly low. However, the pattern correlations again fall quickly after 1 day lead time. The model performed best in three homogeneous regions, as shown in table 6, namely West Central, Central NE, and Northwest regions. The forecast obtained from this deep learning model is comparable to state of the art dynamical models such as provided in Mukhopadhyay et al. (2019) and the PSS from Rao et al. (2019). We found that the ConvLSTM skills for Ld1 and Ld3 are superior or comparable to those in this manuscript for wet and moderate spells.

The forecast skill was also analysed using the ROC curve for homogeneous regions. The ROC analysis was found to be consistent with correlation. We note that the present model reasonably captures the widespread precipitation but still have issues with localized events which might be related to the fact that large scale organized systems have more lifetime and spatial scale which can be captured based on the single variable model attempted here. While the localized extremes are often of short duration and do not have enough memory with them to be taken for the next day when dealing with daily data. Therefore it is still a challenge even for state-of-the-art NWP models to predict such events.

This work is a demonstration of deep machine learning-based algorithms for weather forecasting using only a single variable, which is probably a reason for the steep fall in the efficiency of forecasts after 2 days. However, it is noted that the two day lead predictions of this model compare reasonably against the global forecast system (GFS) T574L64 ($\approx$5 km), adopted from National Centers for Environmental Prediction (NCEP), and tested by the IMD during the 2010s (Durai and Bhowmik 2014). The study reported that areas of negative mean errors spread over most parts of the country from the lead day-2 onwards. With the adoption of higher resolution and improved GFS T1534 ($\approx$ 12.5 km), the efficiency of short range operational forecasts have increased (Mukhopadhyay et al. 2019). It has been reported that GFS T1534 has much improved skill in moderate (15.6 - 64.5 mm day - 1) rainfall categories while there is underestimation for the heavy to very heavy (64.5 - 204.05 mm day - 1) rainfall. Also the extremely heavy rainfall categories are only better on the shorter lead times. This ensemble based state-of-the-art forecasting is efficient but resource intensive and has issues as discussed.

The present model is a proof of concept for a pure AI-based model for short-term rainfall forecasting of the Indian summer monsoon. Despite the fact that the model is fairly simple and only employs one variable, it has given high correlation in several locations. We acknowledge that there may be alternative models that may successfully predict rainfall. We are also experimenting to improve the model. The main purpose of the current study is to introduce an AI model which can be used to forecast monsoon rainfall on short scale.

7. Future work

We observed that the majority of the improvement in the six-variable input model can be accounted for by just two variables (q700 and q850). Thus, the model, in the present form has limitations. Hence, the potential variables which can be used in further studies are sea level pressure, sea surface temperature (SST) and air temperature. The technique can be improved by using more layers in the training and the tuning of hyper parameters.

The custom loss function used for TRMM data can also be experimented on the IMD dataset to improve the data training. Another modification that can be done to handle datasets like IMD with NaN values is to generalize the convolution operator to act on irregular shapes (Vialatte et al. 2016; Pasdeloup et al. 2017). This model has potential to be utilized in short-range forecasting of monsoon precipitation, fire prediction and heat/cold wave forecasting. One can develop a multi-model ensembles using different architectures. Furthermore, there are other models available in the literature including Unet and Transformer (Bojesomo et al. 2021), which can be applied to short range forecasting.

Acknowledgment

The IITM Pune is funded by the Ministry of Earth Science (MoES), the Government of India. This work was done using HPC facilities Pratyush (http://pratyush.tropmet.res.in/) provided by MoES at IITM Pune.

Disclosure statement

The authors confirm that there is no conflict of interest.

Data availability statement

All data, support the findings of this study are available from the corresponding author upon reasonable request.

References

Ballas, N., Yao, L., Pal, C., and Courville, A. (2015). Delving deeper into convolutional networks for learning video representations. *arXiv:1511.06432v4*, pages 1–5.

Barzegar, R., Aalami, M. T., and Adamowski, J. (2021). Coupling a hybrid cnn-lstm deep learning model with a boundary corrected maximal overlap discrete wavelet transform for multiscale lake water level forecasting. *Journal of Hydrology*, 598:126196.

Bojesomo, A., Al-Marzouqi, H., and Liatsis, P. (2021). Spatiotemporal vision transformer for short time weather forecasting. In *2021 IEEE International Conference on Big Data (Big Data)*, pages 5741–5746.

Borah, N., Sahai, A. K., Chattopadhyay, R., Joseph, S., Abhilash, S., and Goswami, B. N. (2013). A self-organizing map–based ensemble forecast system for extended range prediction of active/break cycles of indian summer monsoon. *Journal of Geophysical Research: Atmospheres*, 118(16):9022–9034.

Chattopadhyay, R., Sahai, A., and Goswami, B. N. (2008). Objective identification of nonlinear convectively coupled phases of monsoon intraseasonal oscillation: Implications for prediction. *Journal of the atmospheric sciences*, 65(5):1549–1569.

Chollet, F. (2017). *Deep Learning with Python, 1stedn.* Manning Publications Co., Greenwich, CT, USA.

Dasgupta, P., Metya, A., Naidu, C., Singh, M., and Roxy, M. (2020). Exploring the long-term changes in the madden julian oscillation using machine learning. *Scientific reports*, 10(1):1–13.

Diez-Sierra, J. and del Jesus, M. (2020). Long-term rainfall prediction using atmospheric synoptic patterns in semi-arid climates with statistical and machine learning methods. *Journal of Hydrology*, 586:124789.

Durai, V. and Bhowmik, S. R. (2014). Prediction of indian summer monsoon in short to medium range time scale with high resolution global forecast system (gfs) t574 and t382. *Climate dynamics*, 42(5-6):1527–1551.

Ehsani, M. R., Behrangi, A., Adhikari, A., Song, Y., Huffman, G. J., Adler, R. F., Bolvin, D. T., and Nelkin, E. J. (2021). Assessment of the advanced very high resolution radiometer (avhrr) for snowfall retrieval in high latitudes using cloudsat and machine learning. *Journal of Hydrometeorology*, 22(6):1591 – 1608.

Goswami, B. N., Venugopal, V., Sengupta, D., Madhusoodanan, M., and Xavier, P. K. (2006). Increasing trend of extreme rain events over india in a warming environment. *Science*, 314(5804):1442–1445.

Goswami, B. N. and Xavier, P. K. (2003). Potential predictability and extended range prediction of indian summer monsoon breaks. *Geophysical Research Letters*, 30(18).

Ham, Y.-G., Kim, J.-H., and Luo, J.-J. (2019). Deep learning for multi-year enso forecasts. *Nature*, 573(7775):568–572.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Huffman, G., Bolvin, D., Nelkin, E., Adler, R., et al. (2016). Trmm (tmpa) precipitation l3 1 day 0.25 degree x 0.25 degree v7.

Huffman, G. J., Adler, R. F., Bolvin, D. T., and Nelkin, E. J. (2010). *The TRMM Multi-Satellite Precipitation Analysis (TMPA)*, pages 3–22. Springer Netherlands.

Khan, M. I. and Maity, R. (2020). Hybrid deep learning approach for multi-step-ahead daily rainfall prediction using gcm simulations. *IEEE Access*, 8:52774–52784.

Kim, S., Hong, S., Joh, M., and Song, S.-k. (2017). Deeprain: Convlstm network for precipitation prediction using multichannel radar data. *arXiv preprint arXiv:1711.02316*.

Kothawale, D. and Rajeevan, M. (2017). Monthly, seasonal, annual rainfall time series for all-india, homogeneous regions, meteorological subdivisions: 1871-2016.

Krishnan, R., Singh, M., Vellore, R., and Mujumdar, M. (2020). Progress and prospects in weather and climate modelling. *arXiv preprint arXiv:2011.11353*.

Lara-Benítez, P., Carranza-García, M., and Riquelme, J. C. (2021). An experimental review on deep learning architectures for time series forecasting. *International Journal of Neural Systems*, 31(03):2130001. PMID: 33588711.

Li, W., Gao, X., Hao, Z., and Sun, R. (2022). Using deep learning for precipitation forecasting based on spatio-temporal information: a case study. *Climate Dynamics*, 58(01):443–457. PMID: 33588711.

Manzato, A. (2007). A note on the maximum peirce skill score. *Weather and Forecasting*, 22(5):1148 – 1154.

Marzban, C. (2004). he roc curve and the area under it as performance measures. *Weather and Forecasting*, 19(6):1106–1114.

Moghaddam, M. A., Ferre, P. A. T., Ehsani, M. R., Klakovich, J., and Gupta, H. V. (2021).

Can deep learning extract useful information about energy dissipation and effective hydraulic conductivity from gridded conductivity fields? *Water*, 13(12).

Moghaddam, M. A., Ferre, T. P. A., Chen, X., Chen, K., and Ehsani, M. R. (2022). Application of machine learning methods in inferring surface water groundwater exchanges using high temporal resolution temperature measurements. *arXiv:2201.00726v1*, pages 1–5.

Moon, S.-H., Kim, Y.-H., Lee, Y. H., and Moon, B.-R. (2019). Application of machine learning to an early warning system for very short-term heavy rainfall. *Journal of Hydrology*, 568:1042–1054.

Mukhopadhyay, P., Prasad, V., Krishna, R. P. M., Deshpande, M., Ganai, M., Tirkey, S., Sarkar, S., Goswami, T., Johny, C., Roy, K., et al. (2019). Performance of a very high-resolution global forecast system model (gfs t1534) at 12.5 km over the indian region during the 2016–2017 monsoon seasons. *Journal of Earth System Science*, 128(6):1–18.

Pai, D., Sridhar, L., Rajeevan, M., Sreejith, O., Satbhai, N., and Mukhopadhyay, B. (2014). Development of a new high spatial resolution (0.25× 0.25) long period (1901–2010) daily gridded rainfall data set over india and its comparison with existing data sets over the region. *Mausam*, 65(1):1–18.

Pasdeloup, B., Gripon, V., Vialatte, J.-C., Pastor, D., and Frossard, P. (2017). Convolutional neural networks on irregular domains based on approximate vertex-domain translations. *arXiv preprint arXiv:1710.10035*, pages 1–5.

Pörtner, H.-O., Rober, D., Tignor, M., Poloczanska, E., Mintenbeck, K., Alegria, A., Craig, M., Langsdorf, S., Löschke, S. Möller, V., Okem, A., and Rama, B. (2022). Ipcc, 2022: Climate change 2022: Impacts, adaptation, and vulnerability. contribution of working group ii to the sixth assessment report of the intergovernmental panel on climate change.

Rajeevan, M., Bhate, J., Kale, J., and Lal, B. (2006). High resolution daily gridded rainfall data for the indian region: Analysis of break and active monsoon spells. *Current Science*, pages 296–306.

Rajeevan, M. N. and Santos, J. (2020). India's monsoon mission.

Rao, S. A., Goswami, B., Sahai, A. K., Rajagopal, E., Mukhopadhyay, P., Rajeevan, M., Nayak, S., Rathore, L., Shenoi, S., Ramesh, K., et al. (2019). Monsoon mission: a targeted activity to improve monsoon prediction across scales. *Bulletin of the American Meteorological Society*, 100(12):2509–2532.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., et al. (2019). Deep learning and process understanding for data-driven earth system science. *Nature*, 566(7743):195–204.

Saha, M., Mitra, P., and Nanjundiah, R. S. (2016). Autoencoder-based identification of predictors of indian monsoon. *Meteorology and Atmospheric Physics*, 128(5):613–628.

Saha, M. and Nanjundiah, R. S. (2020). Prediction of the enso and equinoo indices during june–september using a deep learning method. *Meteorological Applications*, 27(1):e1826.

Samadianfard, S., Mikaeili, F., and Prasad, R. (2022). Evaluation of classification and decision trees in predicting daily precipitation occurrences. *Water Supply*, 22(4):3879–3895.

Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., and Woo, W.-c. (2015). Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, pages 802–810.

Shi, X., Gao, Z., Lausen, L., Wang, H., Yeung, D. Y., Wong, W., and Woo, W. (2017). Deep learning for precipitation nowcasting: A benchmark and a new model. 2017. *URL: http://arxiv. org/abs/1706.03458.*

Siami-Namini, S., Tavakoli, N., and Namin, A. S. (2018). A comparison of arima and lstm in forecasting time series. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1394–1401. IEEE.

Singh, B., Krishnan, R., and et al. (2021a). Linkage of water vapor distribution in the lower stratosphere to organized asian summer monsoon convection. *Climate Dynamics*, pages 1709–1731.

Singh, B. B., Singh, M., and Singh, D. (2021b). An overview of climate change science over south asia: Observations, projections and recent advances. *Practices in Regional Science* *and Sustainable Regional Development.*

Vialatte, J.-C., Gripon, V., and Mercier, G. (2016). Generalizing the convolution operator to extend cnns to irregular domains.

Viswanath, S., Saha, M., Mitra, P., and Nanjundiah, R. S. (2019). Deep learning based lstm and seqtoseq models to detect monsoon spells of india. In *International Conference on Computational Science*, pages 204–218. Springer.

Weisstein, E. W. (2020). Statistical correlation. *MathWorld*.

Yin, G., Yoshikane, T., Yamamoto, K., Kubota, T., and Yoshimura, K. (2022). A support vector machine-based method for improving real-time hourly precipitation forecast in japan. *Journal of Hydrology*, 612:128125.

Zeiler, M. D. and Fergus, R. (2014). Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer.

Zhao, P., Wang, Q. J., Wu, W., and Yang, Q. (2022). Extending a joint probability modelling approach for post-processing ensemble precipitation forecasts from numerical weather prediction models. *Journal of Hydrology*, 605:127285.