Sequence Model based Cloudburst Prediction for the Indian State of Uttarakhand

Article *in* Disaster Advances · June 2021

DOI: 10.25303/f2512105

Ananthakrishnan Balasundaram
Vellore Institute of Technology University

Sequence Model based Cloudburst Prediction for the Indian State of Uttarakhand

Sivagami M.[1], Radha P.[1] and Balasundaram A.[2*]
1. School of Computer Science and Engineering, Vellore Institute of Technology (VIT), Chennai, Tamil Nadu, INDIA
2. School of Computer Science and Engineering, Center for Cyber Physical Systems, Vellore Institute of Technology (VIT), Chennai, Tamil Nadu, INDIA
*balasundaram2682@gmail.com

Abstract
Predicting the phenomenon of cloudburst has been a larger than life challenge to many weather and rain scientists. The very nature of cloudburst occurrence itself complicates the prediction of cloudburst. Since, cloudburst downpour occurs over a short span of time and is confined to very narrow geographic location, it is highly difficult for weather scientists to make any cloudburst predictions.

In this work, the authors propose a cloudburst prediction model that leverages deep learning techniques to predict the occurrence of cloudburst in a location. The authors have collected the data pertaining to the cloudburst events that have occurred in the Indian State of Uttarakhand over the past decade and developed the model. Experiments were conducted using time series sequence models namely Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU). Predictive Power Score (PPS) has been used to extract the essential features that are fed as input to these sequence models. The performance of sequence models has been discussed in terms of loss function and accuracy and the results are promising for GRU based model in comparison with other sequence models.

Keywords: Cloudburst, PPS, Deep Learning, GRU, LSTM, Time series sequence models.

Introduction

Cloudburst is a short, sudden, excessive and intense downpour of rain that occurs over a short span of time and over a small area[2]. The formal definition of cloudburst as given by Bernice et al[2] states that cloud burst is a torrential downpour of rain which by its very high intensity and confined area suggests the burst and discharge of a cloud in one single area. The amount of rain water gutting into the geographical area is substantially high and this results in catastrophic events such as flash floods, landslides, damaged roads, loss of human lives, losses to crops, property, cattle etc. Cloudburst phenomenon is mostly observed in hilly regions. In India, most of the cloudburst events occur in the mountainous state of Uttarakhand as listed[8].

Several factors play a pivotal role in the occurrence of a cloudburst event. The prominent factors that influence cloudburst formation may be categorized as geographical parameters, surface parameters, hydrological parameters and type of cloud. Geographical parameters are wind direction, altitude of the place, latitude, ocean currents, wind pressure and velocity etc. Surface parameters include orographic lift, rain shadow, Foehn wind etc. Hydrological parameters include precipitation, evaporation, evapotranspiration, surface water and runoff etc. The type of cloud also has a direct impact in formation of a cloudburst event. Typically, low clouds namely cumulonimbus clouds greatly influence cloudburst formation.

*** Author for Correspondence**

Cloud burst prediction is a challenging task for scientists and researchers mainly because of the nature of the event occurrence. Indian Metrological Department (IMD) states that cloudburst occurs with a high intensity of precipitation which is over 100 mm/h over a very short time period and is confined to a very small region which may be a town or city in a district covering a geographical area of around 20-30 square kilometre[3].

The complexity is further magnified by the galaxy of factors that influence the formation of cloudbursts and culminates in the occurrence of the event. Presently, works are carried out to leverage the services of Doppler radars for predicting in advance the occurrence of any cloudburst event.

But, the cost associated with installing Doppler radars is quite high. An alternate possibility to predict cloudburst occurrence is using machine learning based models that take the various influential factors as input parameters and predict the possibility of cloudburst event occurrence based on the historical training with similar data. This option is cost effective and is easily trainable to enhance the accuracy of prediction compared to deploying hardware components.

This work is focussed towards developing a deep learning model that uses Gated Recurrent Unit (GRU) and Long Short Term Memory (LSTM) networks to predict the possible occurrence of cloudburst event. The dataset used for this work was created with data taken from Fowdur et al[5] for various locations in Uttarakhand where cloudburst events had occurred in the past. PPS has been used to extract the important features from the dataset and these features are fed as input to the time series models.

It must be noted that there is a potential class imbalance scenario with this dataset as 16 or the 20 dataset items are cloudburst events while the remaining are non-cloudburst events. The performance of the model is tested with contemporary works and the results are discussed.

Review of Literature

Tiwari et al[5] proposed Arduino based cloudburst predetermination system based on real time calculation of rainfall intensity. This method is very effective rather than traditional methods of weather forecasting. There is no need for complex assembly as well as it requires less time for prediction. As rain gauges can be built by humans, this method can cost very less.

The proposed method analysed the causes, the time factor and Disaster Risk Reduction measures for HMD and concluded that these disasters cannot be predicted and controlled as it is by nature. However, we could minimise HMD by implementation of pre-disaster mitigation measures by Government and private sector developers with early warning systems. The frequency and intensity of HMD have increased since 1997 leading to disasters in Uttarkashi, Rudraprayag and Chamoli districts.

The major factor which increases HMD is due to geographical factors like human population growth, unstable slope and poor rock conditions. The impact of HMD can be minimised by implementing pre-disaster mitigation measures like public awareness, planning, structural and non-structural measures following disaster mitigation rules by monitoring authorities with early warning systems. We could make their visit to Uttarakhand safe and peaceful by providing proper guidelines to tourists, pilgrims, mountaineers with the help of State agencies. This could be done by the help of NGOS, E-media, news and weather charts etc.

The cloudburst event in Himalayan region was examined in the early hours of 16 July 2003[4]. Although the storm lasted even less than 30 minutes, followed by flash floods which affected 100's of people. In order to predict a cloudburst event with cloud microphysics parameterization and horizontal resolution, fidelity of MM5 configured with multiple nested domains was examined. This MM5 model predicts the rainfall 24hrs before hand.

As per the analysis of hydrometer structure, MM5 captures the salient features that are observed by the TRMM satellite. It generates cloudburst events at 30Km grid resolution which is too early. In order to improve the cloudburst event, the experiment was carried out at 9km and 3Km grid-resolution. MM5 simulates the convective activity, shear and moisture build-up as per the analysis of rainfall over Himachal Pradesh.

The MM5 overestimates hydrometer content as and when compared between simulated vertical structure of hydrometer and TRMM due to higher vertical and horizontal resolution of MM5. Also, inter-comparison of two moisture schemes produces high amounts of rainfall. We would need high resolution measurements for an accurate description system and its development and decay would help to evaluate simulation results. The interactions among deep convection, orography and cloud-microphysics are to be quantified. Also, the numerical realization will improve the cloudburst prediction.

The work done towards estimating the outflow flood hydrograph[7] at the outlet of an ungauged Himalayan catchment using Synthetic Unit Hydrograph is noted. They highlighted the need for a comprehensive database for effective disaster management over the entire Himalayan region and it was concluded that cloudburst occurs due to flood disasters in hilly areas. SCS method is used for rough estimations and Synthetic's method can be used for estimation of flood hydrograph characteristics using corrected values of coefficients.

In another work, the authors examined that cloudburst triggered natural hazards[12] in Uttarakhand Himalaya's are due to flash floods and landslides. In order to prevent disasters, we would need to avoid construction of human settlements, institutions and infrastructural facilities with seasonal streams. Also, tree plantation on degraded land will reduce hazards.

Sajwan et al[10] focussed on significant aspect of cloudburst and respective hazards in Uttarakhand. Cloudburst and anthropogenic activities play a prominent role in triggering flash floods and landslides. They concluded that different aspects of geomorphology and climatology would help in development of cloudburst forecast models and according to their studies, we would need extreme precipitation events and respective disasters among the dwellers of the area. Samya et al[11] proposed the forecasting cloudburst techniques using Data mining as well as Artificial Neural Network (ANN) and they had observed that Data Mining Technique, ANN, Fuzzy logic and ANFIS result in better accuracy.

Several data mining techniques[1] based on machine learning and conventional methods for a rainfall forecast have been provided. Also, these techniques were used to generate prediction models based on historical data of Tamil Nadu Govt. website. As farmers face insecurity in their business due to multiple factors. Rainfall plays a prominent role in the production system and in financial returns. So, for digging the information of month to month rainfall, they had used Multiple Linear Regression and it was evaluated that decision tree and K-means clustering are the best suited data mining techniques with increase in size of training dataset, the accuracy is increased.

Another work[5] focussed on giving a climatic profile. Mauritius faced the issues due to climatic conditions that could not be predicted. So by using detailed analysis of techniques, a real-time weather forecasting system has been proposed. Also, a framework for real-time cloud-based forecasting has been set up in Mauritius. The proposed system uses IBM Bluemix cloud platform for predictive analytics. It has the potential to support emergency managers like fire services, SMF and the general public as well.

Methods

The proposed approach uses two time series sequence models namely GRU and LSTM to predict cloudburst event occurrence. Figure 1 shows the block diagram of the proposed model. The functioning of each block is discussed in detail.

Feature Selection: Once we take any dataset, we might find the best features for better predictions. In order to build a machine learning model and to get required predictions, we would need to choose the best features. Correlation is the statistical summary of the relationship between two variables and to find how close these variables are in linear relationship with each other. If two variables are linearly dependent, then we could say that those two variables are highly correlated with each other and vice versa. If two variables are non-linearly dependent, then we could say that those two variables are highly correlated with each other. When two features are highly correlated and linearly dependent, they would be having the same effect on dependent variables. Hence, we can drop one feature out of the two. Null hypothesis is a statistical statement which shows that there is no significant relationship between two variables.

P-value: P-value is a probability value of a statistical model which gives the probability of an observation under the assumption that a particular hypothesis is true. This probability can be used to reject or accept hypothesis. If p-value is higher than the significant level, we accept the null hypothesis. Else, we reject the null hypothesis. We could quantify the relationship between two variables with correlation statistics. If there is linear association between the two variables, then they are correlated. Else, the variables are uncorrelated.

We could classify the correlation variables as per the type of correlation as well. If two variables are linearly correlated then it is called positive correlation. Else, it is a negative correlation. If there is no relationship between two variables then we could say zero correlation or uncorrelated.

Pearson Correlation Coefficient: A general correlation statistic used for continuous variables is Pearson correlation coefficient and the value lies between -1 to 1. If there is no linear association between two variables, then the value would be 0. If it is linearly correlated, then the value would be greater than 0. Else, the value would be less than 0. The correlation between two variables can be visualized by plotting scatter plot of the data. A residual plot exhibits if the dataset follows a random pattern or not. Then it would be possible to transform the raw data to linear. Transformations can often fit between X (variable) and Y (variable). While applying transformation to the raw data, the plot becomes linear.

Predictive Power Score (PPS): For solving data of science problems, we would need to extract the insights of data, which involves finding the relationship between two features and correlation matrix is most widely used to accomplish this purpose. However, for a few instances, correlation matrix is unable to convey the insights of data. In order to overcome this issue, we would be using Predictive Power Score. It can work with non-linear relations, categorical data and nominal data as well, in addition with numerical values. Correlation matrix works with only numerical values. PPS is asymmetric that means if column A predicts column B, that does not mean B can also predict A. Generally, PPS value lies between 0 and 1. If the objective is numerical, then we could use the Regression Decision tree and calculate Mean Absolute Error (MAE). If the objective is categorical, then we could use the Classification Decision Tree and calculate the weighted F1.

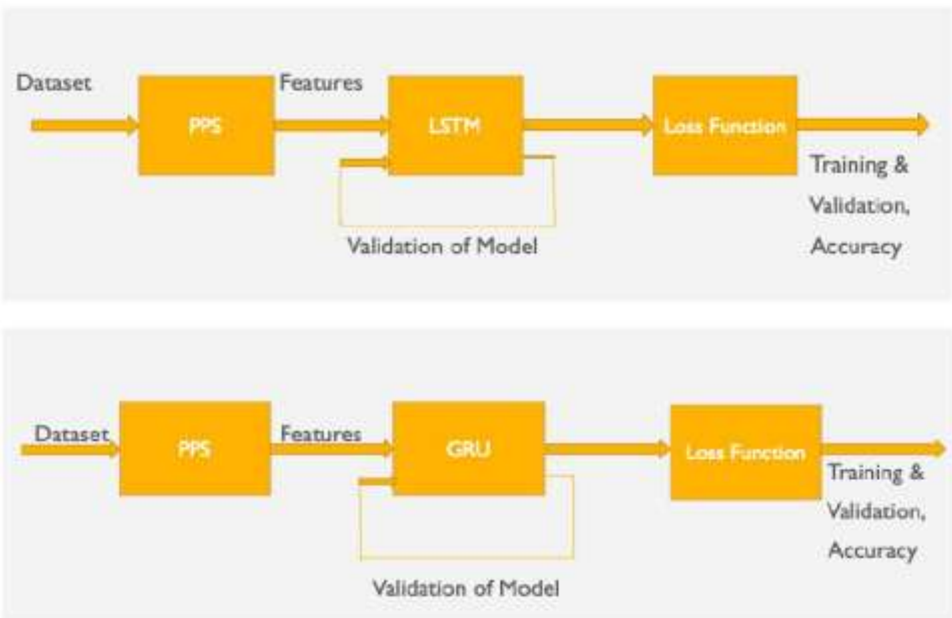

Fig. 1: Block diagram of the proposed model

PPS detects the non-linear relationships. PPS has been used to identify the nonlinear relationship between independent variables to dependent variables. Then, by applying PPS we got the best 41 features out of 52. With the trial and error method, we have obtained values >0.5 and the features of the same are then fed into the LSTM and GRU model. If PPS score is high, then that would predict Y pretty good using X. A low PPS score would imply that Y is hard to predict using X. The PPS acts as a correlation coefficient, but it is different in many ways like it is asymmetric.

Also, it can summarise the predictive value among categorical variables as well as nominal data. The weather dataset is given as input to the Predictive Power Score. We got the best features out of PPS. The best features were fed into the LSTM model. LSTM is an evolution on a RNN. Normal RNN modules took output of a last layer over a single TANH function. LSTMs use feedback loops and gates to remember. LSTMs have 4 NNs layers in each module.

In general, for deriving the relationship between features, we would use correlation matrix. With the use of this correlation matrix, we could find the relationship between nominal values. However, it will not be possible to identify the relationship between non-linear features, categorical variables. Hence, we chose to use Predictive Power Score. We applied predictive power score technique and based on the relationship between features, we identified the best features out of it whose p-value is greater than 0.5. We had totally 47 features in the weather dataset. While implementing predictive power score technique, we identified the best features of 3 attributes named as Weather hourly WeatherCode, Weather hourly humidity and Weather hourly Cloudcover based on the relationship between features whose p-value > 0.5. Also, for classification purposes, we would be using the LSTM and GRU model and predicting cloudburst.

Deep Learning Model Consideration

Long Short Term Memory (LSTM): As RNN has a short-term memory, there is an issue of Long-Term Dependencies. So, it does not seem to be able to learn in case the gap between the relevant information and the point where it is needed becomes very large. This issue has been resolved in Long Short-Term Memory networks called simply LSTMs.

Emerging technologies often lead to the development of new Deep Learning Artificial Neural Networks (ANNs). Such is the case with Convolutional Neural Networks (CNNs) and Long Short-Term Memory Networks (LSTMs). As LSTM contains internal memory, it can enable RNN to remember the inputs over a long period of time. The LSTM can indeed perform read, write and delete information from memory.

A basic LSTM network shown in figure 2 consists of memory blocks called as cells, input gate, forget gate and output gate. The cell remembers the values over random time intervals and the three gates manages flow of information into and out of the cell. The purpose of input gate is to allow information to the cells. The forget gate deletes the information which is not necessary for the accomplishment of the respective task. This stage plays a vital role in order to achieve an optimizing performance of the network.

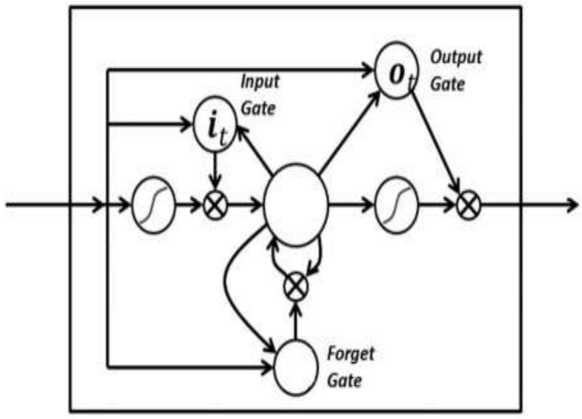

Fig. 2: Typical LSTM network[6]

The output gate selects and outputs the required information. Gates in LSTM are of sigmoid activations and range from zero to one. This is helpful to update or forget data if any number multiplied by 0 is 0 that shows that values will disappear or forgotten. In case the number is multiplied by 1, then it retains the same value, hence, the value will be the same. The network can learn the data and keep it which is important and forgotten which is not important. This is how internal memory acts as a gated cell and works to accomplish tasks.

The issues of vanishing gradients are solved by using LSTM as it keeps the gradients steep enough that keeps the training relatively short and the accuracy high. LSTMs can be applied to Deep Learning tasks which include prediction on sequential and time series data.

Gated Recurrent Unit (GRU): GRU is one kind of recurrent neural networks which is similar to LSTM but comprises of only two gates i.e. update gate and reset gate. The two gates decide what information has to be passed to the output. The GRU captures dependencies that exist in large sequences of data without removing the information from previous parts of the actual sequence. We could solve the issues of vanishing gradients and exploding gradients through gated units. This gated structure uses to manage and control the flow of contents across cells in the neural network.

The sequential data along with internal memory or hidden state are being fed to the GRU cell at each step of time period, this process resembles like a relay mechanism and eventually, it gives required output. The hidden state could hold long-term and short-term dependencies as well.

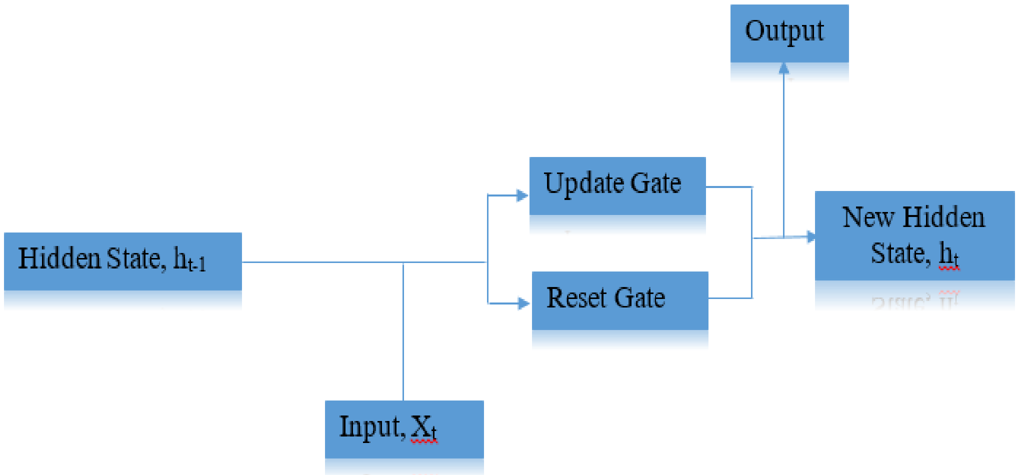

Fig. 3: Structure of a GRU cell

The gates in GRUs can be trained faster and filters out the unnecessary information while keeping only useful contents. These gates or vectors range from 0 to 1 that get multiplied with sequence input data and the hidden state or internal memory. A value of 0 in the gated output resembles the corresponding sequence input data or hidden state is not important and hence, it returns to 0. A value of 1 in the gated vector denotes that the respective input or hidden state is important and which would be used. The sequential input data and hidden state both will be multiplied by their respective weights and summing them prior to passing through sigmoid active function to transform values between 0 and 1. This reset gate is mainly used to filter out the unimportant information and retains the data which is useful.

The update gate is derived and calculated from sequential input data and hidden state of previous time step and at the current time step input. The update gate is derived and calculated from sequential input data and hidden state of previous time step and at the current time step input. These inputs are multiplied with respective unique weights, summed up and then fed into a sigmoid function in order to transform the values from 0 and 1. The update gate helps our model to derive how much previous information need to be passed through the output, which decides to remove the risk of vanishing gradient issue.

$$gate_{update}=\sigma(W_{inputupdate}\cdot x_t+W_{hiddenupdate}\cdot h_{t-1})$$

The update gate will be multiplied with the previous hidden state in order to get final output.

$$u=gate_{update}\odot h_{t-1}$$

Eventually, the inverse version of update gate and reset output gate will be multiplied in order to know the new information stored in hidden state and the result will be summed up with update gate. Finally, we will get the resultant updated hidden state.

$$h_t=r\odot(1-gate_{update})+u$$

Results and Discussion

We created the data in the form of a sine wave with time period for visualization in order to model many oscillations for the LSTM network to train the dataset. This is depicted in figure 4. The weather dataset that had been used is split as 70% for training and 30% for testing. The data is learnt from the sine wave by LSTM model which will then be used to predict the next n-steps in the series. We transform the data and load the dataset from CSV or excel file to the numpy array that would feed to LSTM network.

Then the dataset is loaded, trained and finally the network is built. For building the model, we used a sequential function and network structure of 1, 50, 100, 1 where one input layer is kept that feeds into LSTM layer of 50 neurons, that in turn feeds into another LSTM layer of 100 neurons which is then connected to 1 neuron of fully connected to normal layer of 1 neuron with linear activation function that will give prediction for next time step.

A total of 10 training epochs were used in this LSTM model that would cycle through all sequence windows in the training set once. The data is then split into training and testing set. In order to predict multiple folds, the first window from testing data is used as an initiation. For each time step, we pop the oldest entry out of the window and add the prediction to the next time step to the front of the window. The window is shifted so that it slowly builds the predictions till the window is fully having predicted values.

The model was initially tested by taking into consideration all features as such from the dataset and then with the inputs received from PPS. The performance of both these approaches has been discussed in terms of loss function and classification accuracy for two time series sequence models namely LSTM and GRU.

Experimental results without PPS for LSTM: Figure 5 shows the loss and classification accuracy for LSTM model without PPS. X axis denotes the number of epocs while the Y axis denotes the accuracy and loss values respectively.

The loss was estimated at 0.463 while the classification accuracy was 0.925. Initially the training accuracy at one epoch was 65% and the accuracy was increased to 77% from second epoch till seven epochs. It could also be observed that the training accuracy has been increased linearly up to 92% from epochs seven to ten.

Similarly, the validation accuracy for first epoch was 77% and it continued to increase linearly up to 92% till the tenth epoch as shown in the figure. It could easily be inferred that the validation accuracy increased with increasing the number of epochs. Also, the loss decreased with the increase in epochs.

Experimental results without PPS for GRU: Figure 6 shows the loss and classification accuracy for GRU model without PPS. X axis denotes the number of epocs while Y axis denotes the accuracy and loss values respectively. The loss was estimated at 0.363 while the classification accuracy was 0.933. Initially the training accuracy at first epoch was 30% and the accuracy has been increased to 83% after four epochs and then slightly decreased at the fifth epoch. From there, the accuracy has been increased linearly up to 93% for ten epochs. Likewise, the validation accuracy at first epoch was 75% with the same accuracy upto 5 epochs and then increased linearly up to 93% at sixth epoch and continued till ten epochs as mentioned in figure 6. It can be inferred that accuracy increases with increasing epochs.

Experimental results with PPS for LSTM: Figure 7 shows the loss and classification accuracy for LSTM model with PPS. X axis denotes the number of epocs while the Y axis denotes the accuracy and loss values respectively. The loss was estimated at 0.226 while the classification accuracy was 0.951. X axis denotes the number of epocs while Y axis denotes the accuracy and loss values respectively. It can be observed that the loss is less and accuracy is more when compared to LSTM without PPS scenario.

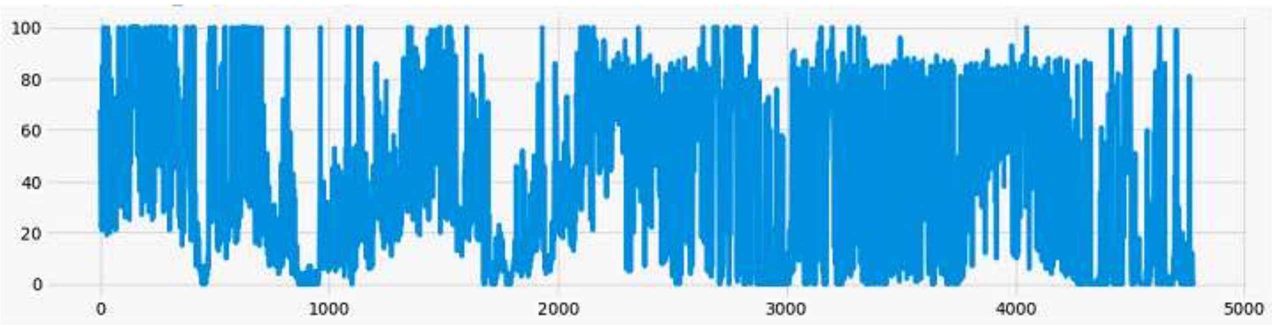

Fig. 4: Visualization of data in sine wave form

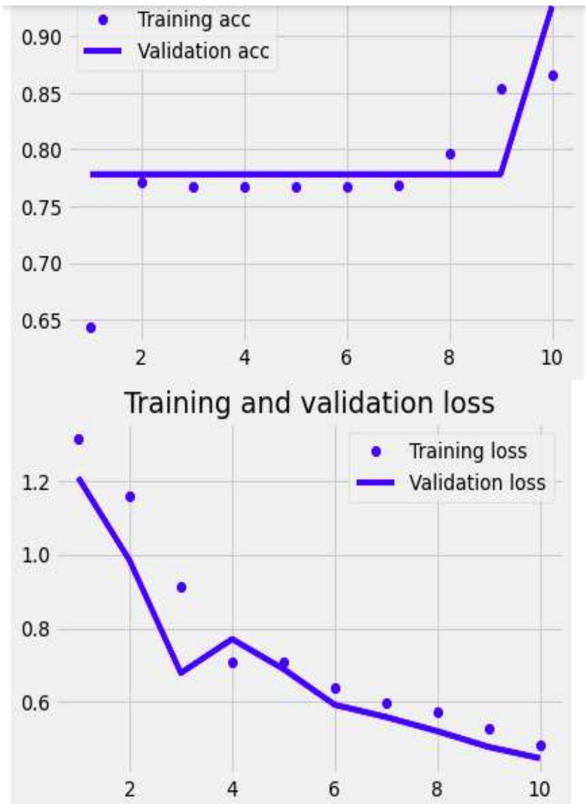

Fig. 5: Loss and classification accuracy for LSTM model without PPS

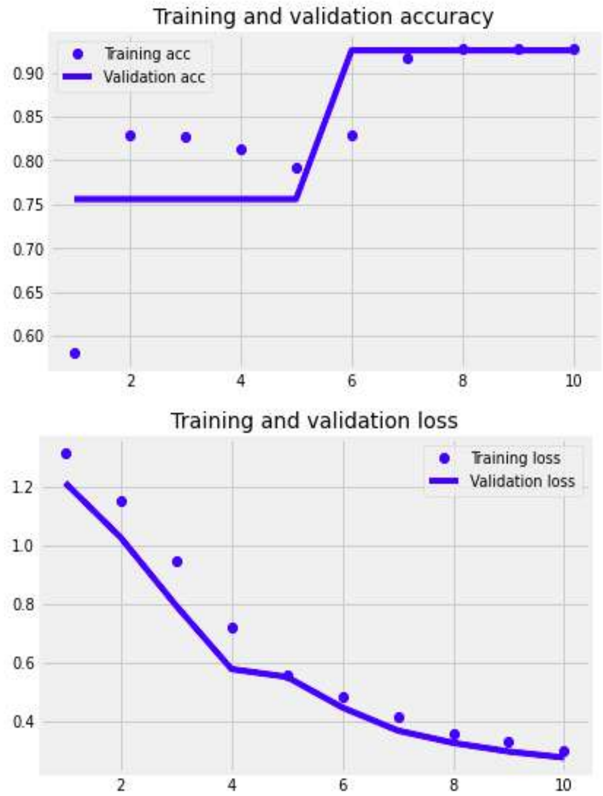

Fig. 6: Loss and classification accuracy for GRU model without PPS

Fig. 7: Loss and classification accuracy for GRU model without PPS

Experimental results with PPS for GRU: Figure 8 shows the loss and classification accuracy for GRU model with PPS. X axis denotes the number of epocs while the Y axis denotes the accuracy and loss values respectively. The loss was estimated at 0.359 while the classification accuracy was 0.921. X axis denotes the number of epocs while the Y axis denotes the accuracy and loss values respectively. It can be observed that the loss is less when compared to GRU without PPS scenario.

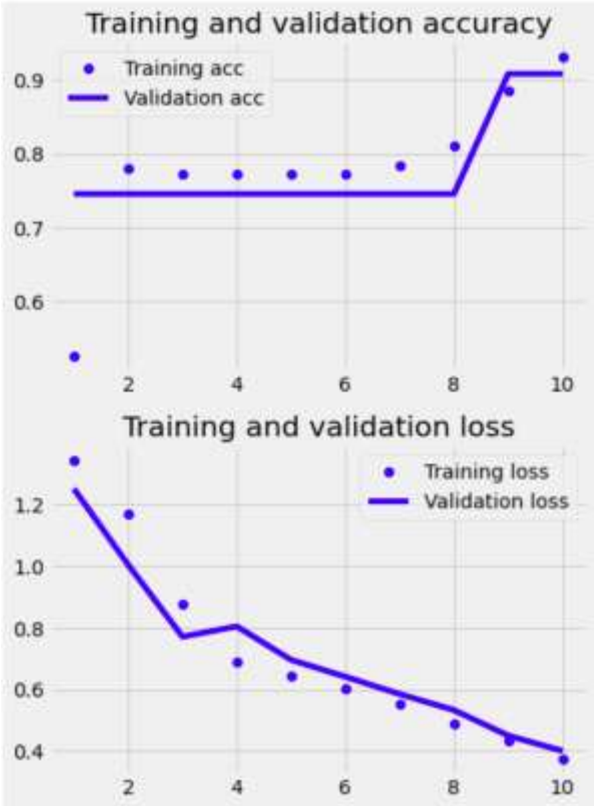

Fig. 8: Loss and classification accuracy for GRU model with PPS

Conclusion and Future Work

This work focussed towards developing a model that could predict extreme rainfall events such as cloudburst. As part of this work, a dataset comprising of cloudburst events that had occurred between the years 2010 and 2020 in the Himalayan state of Uttarakhand was created. The dataset comprised of 16 cloudburst events. Like in many extreme event dataset collections, this dataset comprising of cloudburst events also has a certain degree of class imbalance. Then, the proposed time sequence model was developed using two algorithms namely LSTM and GRU. To enhance the feature extraction, PPS was used and it considered relevant features that would aid better classification results. The experiments were conducted for both LSTM and GRU models with and without PPS usage. The results indicated that using PPS helped in reducing the loss value for both LSTM and GRU models.

Future work will be directed towards expanding the feature set in an attempt to gather more insight as input to the model. Also, efforts will be directed towards developing an expanded dataset comprising of such cloudburst events across the globe and in turn working towards enhancing the model for greater accuracy in terms of predicting cloudburst events.

References

1. Aswini R., Kamali D., Jayalakshmi S. and Rajesh R., Predicting Rainfall and Forecast Weather Sensitivity using Data Mining Techniques, *International Journal of Pure and Applied Mathematics*, **119(14)**, 843-847 **(2018)**

2. Bernice Rosenzweig, Ruddell Benjamin L., Lauren McPhillips, Robert Hobbins, Timon McPhearson, Zhongqi Cheng, Heejun Chang and Yeowon Kim, Developing knowledge systems for urban resilience to cloudburst rain events, *Environmental Science & Policy*, **99(1)**, 150-159 **(2019)**

3. Das P.K., Prafullanagar PO- Global warming, glacial lakes and cloud burst events in Garhwal –Kumaon Himalaya, A hypothetical analysis, Glob Warm 5-12 **(2015)**

4. Das S., Ashrit R. and Moncrieff M.W., Simulation of a Himalayan cloudburst event, *Journal of Earth System Science*, **115(3)**, 299-313 **(2006)**

5. Fowdur T. et al, A Framework for a Real-Time Cloud-Based Weather Forecasting System for Mauritius, *International Journal of Mechatronics, Electrical and Computer Technology (IJMEC)*, **7(1)**, 3563-3581 **(2017)**

6. https://builtin.com/data-science/recurrent-neural-networks-and-lstm **(2021)**

7. Kansal M.L. and Thakur A., Flood Estimation for a Cloudburst Event in an Ungauged Western Himalayan Catchment, *Int J Hydro*, **1(6)**, 1-30 **(2017)**

8. Khanduri Sushil, Cloudbursts Over Indian Sub-continent of Uttarakhand Himalaya: A Traditional Habitation Input from Bansoli, District-Chamoli, India **(2020)**

9. Pandey V.K. and Mishra A., Causes and Disaster Risk Reduction Measures for Hydrometerological Disaster in Uttarakhand, India: An Overview, *International Journal of Current Research in Science and Technology*, **1(3)**, 61-80 **(2015)**

10. Sajwan K.S., Khanduri S. and Bhaisora B., Some significant aspect of cloudburst with especial reference to devastating landslides at Bastari, Naulra and Didihat region, Pithoragarh district, Kumaon Himalaya, Uttarakhand. *Int. J. Current Res.*, **9(7)**, 54255-54262 **(2017)**

11. Samya R. and Rathipriya R., Predictive analysis for weather prediction using data mining with ANN: A study, *International Journal of Computational Intelligence and Informatics*, **6(2)**, 149-153 **(2016)**

12. Sati V.P., Cloudburst triggered natural hazards in Uttarakhand Himalaya: mechanism, prevention and mitigation, *Int J Geol Environ Eng*, **12(1)**, 45-38 **(2018)**

13. Tiwari A. and Verma S.K., Cloudburst Predetermination System, *IOSR Journal of Computer Engineering (IOSR-JCE)*, **17(3)**, 47-56 **(2015)**

14. Woolley R.R., Marsell R.E. and Grover N.C., Cloudburst Floods in Utah, Water Supply Paper No. 994, US Geological Survey, Washington, DC, USA 1850-1938 **(1946)**.

(Received 11th April 2021, accepted 16th May 2021)