A Light weighted Dense and Tree structured simple recurrent unit (LDTSRU) for flood prediction using meteorological variables

Arun Mozhi Selvi Sundarapandi[1], Sundara Rajulu Navaneethakrishnan[2], Hemlathadhevi A[3], Surendran Rajendran[4,*]

[1] Department of Computer Science and Engineering, Holycross Engineering College, Thoothukudi, 628851, India. arunmozhiselvisundarapandi@gmail.com

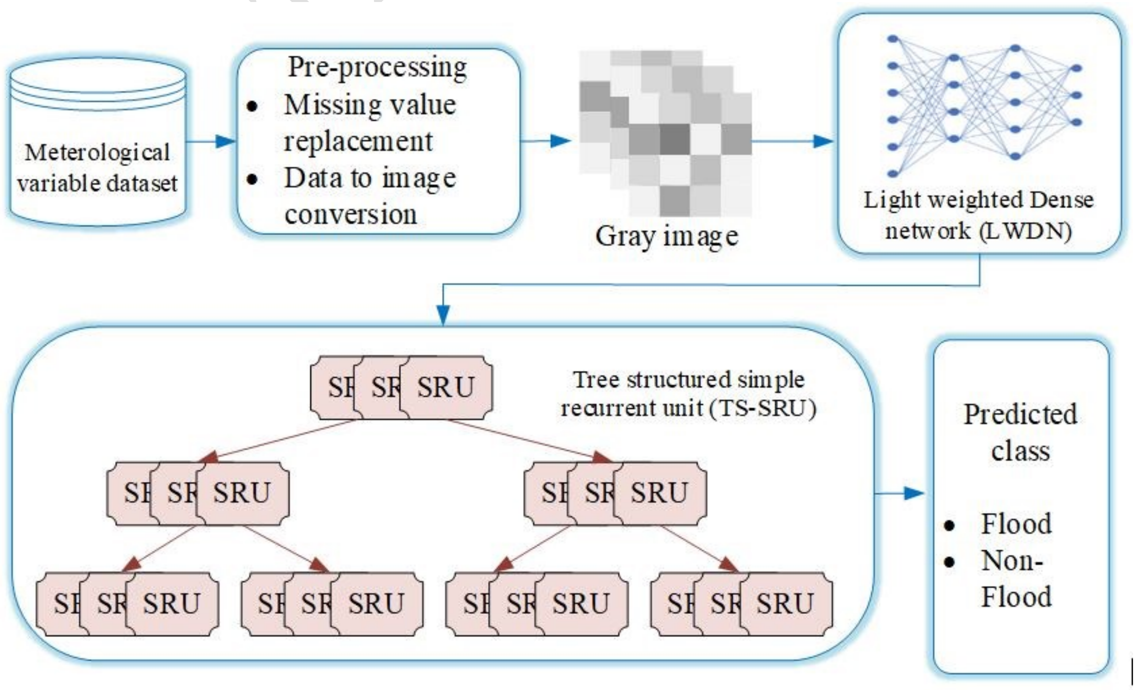
[2] Department of Computer Science and Engineering, Dhanalakshmi Srinivasan University Trichy, Samayapuram Campus, Tiruchirappalli, India. drsundararajulu@gmail.com

[3] Department of Computer Science and Engineering, Panimalar Engineering College, Chennai, 600123, India. hemlathadhevi@gmail.com

[4] Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, 602105, India.

[*]Corresponding author: surendranr.sse@saveetha.com

GRAPHICAL ABSTRACT



ABSTRACT

Floods cause significant harm around the world each year. Predicting floods accurately and in a timely manner can greatly reduce the loss of human life and property. Thus far, a number of modelling approaches have been described for automatic flood detection; these approaches primarily capture temporal dependences while ignoring patterns related to relative humidity, wind speed, and rainfall intensity—all of which are critical for flood prediction. This paper presents a novel prediction method by combining a Light weighted Dense network and Tree structural simple recurrent unit (LDTSRU). First, a light-weighted dense network is used to convert the input meteorological variables into grayscale images and identify any remarkable patterns between the variables. The nonlinear relationship between the input and output data is then automatically learned using the Tree structural simple recurrent unit (TS-SRU). It is also capable of efficiently comprehending the order and flow of events that culminate in a flood. The public flood detection dataset is used to confirm the accuracy of the proposed model by comparing it with state-of-the-art methods. According to experimental findings, LDTSRU can perform better with less training time. LDTSRU has attained 2.53% higher average accuracy and better average precision and average recall compared to well-known state-of-the-art techniques.

Keywords: Flood detection, meterological parameter, Tree structure, and Light weighted dense network, simple recurrent unit.

1. Introduction

One of the most common and destructive natural events that can occur anywhere in the world is flooding. Many people consider the possibility of flooding to be the most common kind of natural disaster. Saravanan et al. (2023) discussed about how the Floods occur frequently in all regions of the world. But they differ from place to place in terms of their traits and strength. Munawar et al. (2022) produce the result of approximately 84% of all-natural disaster-related fatalities worldwide are attributed to floods. Panahi et al. (2021) stated floods are common in many nations, resulting in direct economic losses of US$ 60 billion as well as hundreds of fatalities and injuries each year. Wang et al. (2024) explained one of the most crucial non-engineering approaches to flood control and disaster reduction is flood forecasting. Jin et al. (2024) and Shivarudrappa et al. (2023) determined flood is becoming a hot topic in multidisciplinary study and plays important roles in disaster monitoring, risk identification, forecasting, and warning.

Chen et al. (2022) and Yu et al. (2024) access the numerous elements, including precipitation, evaporation, solar radiation, underlying surfaces, and air circulation, frequently have an impact on floods. Gurbuz et al. (2024) exhibits considerable nonlinearity and high uncertainty as composite attributes. Aljohani et al. (2023) executes the flood forecasting has never been an easy process because of the intricacy of the available data. Conventionally, flood predictions have been made by the creation of distributed hydrologic models based on both physics and empirical data. Rasheed et al. (2022) stated large computing resources and high-resolution geographical data with respect to catchment features and initial boundary conditions are specifically needed for the physics-based models. These reasons make physics-based models unsuitable for large-scale research and practical applications.

Hasan et al. (2023) discussed to include machine learning models while researching and evaluating flood susceptibility. Costache et al. (2024), Mehravar et al. (2023), Aiyelokun et al. (2023), and Fang et al. (2021) discussed various Machine learning techniques have been broadly applied to forecast floods using variables like as temperature, water velocity, humidity, precipitation, and level. Machine learning methods including decision trees (DT), support vector machine (SVM), random forest (RF) and artificial neural network (ANN) have shown strong characteristics and produced positive outcomes for flood prediction in recent years. Using these methods, flood modelling is usually presented as a supervised learning problem, where rules and latent associations of data are obtained by labelling both inputs and outputs, which then predicts the labels of unobserved data after learning. However, traditional machine learning models may not perform well as dataset sizes grow and may not scale effectively with large datasets.

Recent advances in graphic processing units (GPU) and artificial intelligence (AI) have made deep learning applications and creative methods based on multilayer artificial neural networks (ANN) possible. Deep learning models have been successfully applied to a variety of real-world scenarios, particularly time series prediction. Luppichini et al. (2022) explained one of the most well-liked, effective, and extensively utilized learning methods is the recurrent neural network (RNN), which is frequently used in flood prediction studies. An RNN can analyse sequence data because of the connections, which form a loop between its units. However, long-term dependencies are an intrinsic problem of ordinary RNN. This has been tackled by certain specialized RNN types, including Long Short-Term Memory (LSTM) by Wei et al. (2020). However, temporal dependence of state computations affects scaling recurrent networks, like LSTM and gated recurrent unit (GRU). It means that the computation of every step is halted until the preceding phase is fully executed.

Recurrent networks are slower than other models because of this sequential dependency, which also restricts their ability of parallelism by Ling et al. (2021). Simple Recurrent Units (SRU) is a recently proposed enhanced version of RNN, which abridge calculations and reveal more parallelism for reducing the training time. Khan et al. (2022), Kimura et al. (2019), and Chen et al. (2021) researchers have attempted to forecast the flood using the automatic feature extraction capability of CNN. But they are unable to comprehend the relationship between sequence information and process sequence data. These points motivate us to integrate CNN with SRU model for flood detection. They are unable to comprehend the relationship between sequence information and process sequence data. These arguments encourage us to combine the SRU model and CNN for flood detection. The main contributions of this research work are listed as follows:

- To predict floods based on different meteorological variables including temperature, rainfall, Relative Humidity, Wind Speed, Cloud Coverage, Bright Sunshine, and so on.
- To improve the flood detecting accuracy by integrating Light weighted Dense network and SRU.
- To capture the patterns related to different meteorological variables and increases the features of the input data using Light weighted Dense network.
- To introduce a tree structured SRU based network to capture non-linear relationships between features and the target variable. By using non-linear activation functions, parallel processing capabilities, and a hierarchical structure, the Tree Structured SRU architecture is able to capture non-linear correlations between meteorological factors and flood occurrence. When it comes to managing temporal and spatial dependencies, increasing resilience and generalisation, increasing efficiency and parallelism, and managing modelling complexity, SRUs are much superior to linear or shallow models. Their unique characteristics render them especially appropriate for the intricate and ever-changing task of forecasting floods.
- To verify the effectiveness of the proposed LDTSRU by comparing it with state-of-the-art methods on publicly available dataset.

The rest of the paper is structured as follows: Section 2 reviews the most recent papers of flood detection. Section 3 explains the proposed detection framework in details. Section 4 validates the efficacy of proposed model through simulation. Section 5 stated the conclusion.

2. Literature survey

One of the deadliest natural disasters in India's coastal regions is flooding. Sawaf et al. (2021) utilized data-driven framework to identify the latent correlations between several hydrological variables during floods. Also, they sought to determine whether data-driven algorithms could extract disastrous flood records outside of the training area by analyzing the internal properties of training inputs. They created a two-layered RNN model to capture the hidden correlations between the inputs in order to accomplish these goals, and they used quantitative and qualitative analysis to look into the model's prediction power.

Balamurugan et al. (2022) applied machine learning (ML) techniques to create an efficient flood prediction system that would assist to minimize the loss of life and property. The ML models have been created using decision trees (DTs), random forests (RFs), support vector machines (SVMs), and k-nearest neighbors (KNNs). A stacking classifier were employed to address the oversampling and low accurateness issues. Kaur et a. (2022) presented an energy efficient cloud-aided flood prediction framework. The system makes use of a Bayesian belief network (BBN) for the identification of flood events. For flood monitoring and forecasting, the cloud layer leveraged the optimization-based adaptive neuro fuzzy inference system (ANFIS) as a vulnerability analysis component.

Singh et al. (2023) examined the effects of dangerous precipitation and geometric parameters on the hydrologic reactions and river flooding sensitivity of the Brahmani River. The digital elevation model was used to compute geometric variables and detect variation in the radical precipitation files using precipitation data from 1991 to 2021. Moreover, the wet cycle frequency on varying period of 1, 3, 12, and 24 months and their relationship to river floods were determined using the Standardized Precipitation Index (SPI). Du et al. (2021) utilized soil brightness temperatures to asses the flood across southeast Africa at the time of cyclone. The forecasting system has been developed by selecting categorization and regression tree framework with Landsat data. Gharakhanlou et al. (2023) considered different climatological, hydraulic and geospatial parameters for assessing the flood susceptibility. The strength of correlations between the variables was asses using multicollinearity analysis. The Quantile categorization approach was used to reclassify flood training features, and the Frequency Ratio (FR) method was used to assess the relative relevance of each class within a given flood training feature. Also, Spearman correlation analysis was used to examine the spatial link among the flood training features and the flood susceptibility map.

Letessier et al. (2023) presented a new ML technique called the Adaptive Structure of the Group Method of Data Handling, which combines air temperature and precipitation data with watershed characteristics to predict daily river flow rates. High accuracy was attained by the most basic model, which only has three parameters: maximum temperature, precipitation, and historical daily river flow discharge. John et al. (2023) predicted the flood using an enhanced PCA and a one-dimensional CNN. This framework was trained utilizing the day-to-day and monthly rainfall intensity of a state. At first, a linear unsupervised statistical transformation and an enhanced PCA were employed as feature selection techniques for handling high dimensionality data. A 1D-CNN framework predicted the flood according to the selected features.

This review explains that knowledge and ability to forecast flood events are mostly dependent on meteorological factors. Precise monitoring and modelling of these variables allow improved forecasts, prompt warnings, and efficient flood management. This analysis also shows that both RNNs and CNNs have their unique advantages and disadvantages when it comes to flood detection. CNNs are excellent at efficiently extracting features, whereas RNNs are good at capturing temporal dependencies. However, RNNs are hard to parallelize, which restricts their flexibility and lengthens the training period. On the other hand, SRUs can process and train data more quickly because of their design, which makes it easier to parallelize. Thus, the proposed model integrating the lightweight dense network with SRUs and introducing tree structure to better capture long-term dependencies by aggregating data in a hierarchical fashion

3. The Proposed Method

Flood forecasting has never been an easy process because of the intricacy of the available data. Floods can be predicted with great accuracy using ML approaches based on characteristics including temperature, humidity, rainfall, wind speed, and cloud cover. This study proposes a novel Light Weighted Dense and Tree Structural Simple Recurrent Unit (LDTSRU) for flood forecasting. Figure 1 summarizes the proposed flood detection model, which includes the pre-processing step, feature map creation, and classification model. First, flood prediction datasets provide the source data. After that, the input data such as minimum temperature, maximum temperature, rainfall, Relative Humidity, Wind Speed, Cloud Coverage, Bright Sunshine, period and Altitude is pre-processed through the replacement of the missing values with any arbitrary value from the same column. Next, they will be converted into a 2-dimensional format in order to prepare each sample for classifier. Then, this image is given as input to our proposed LDTSRU to predict the flood. Light weighted dense network is used in the proposed LDTSRU to create feature maps from grayscale images. The class will then be predicted using a Tree structural simple recurrent unit (TS-SRU). LDTSRU, with its hierarchical and dense network capabilities, is well-suited to offer comprehensive insights into the dynamics behind flood predictions. One can increase prediction accuracy and preparation and obtain a deeper understanding of the mechanisms causing floods by evaluating the learnt patterns and features using a variety of ways.

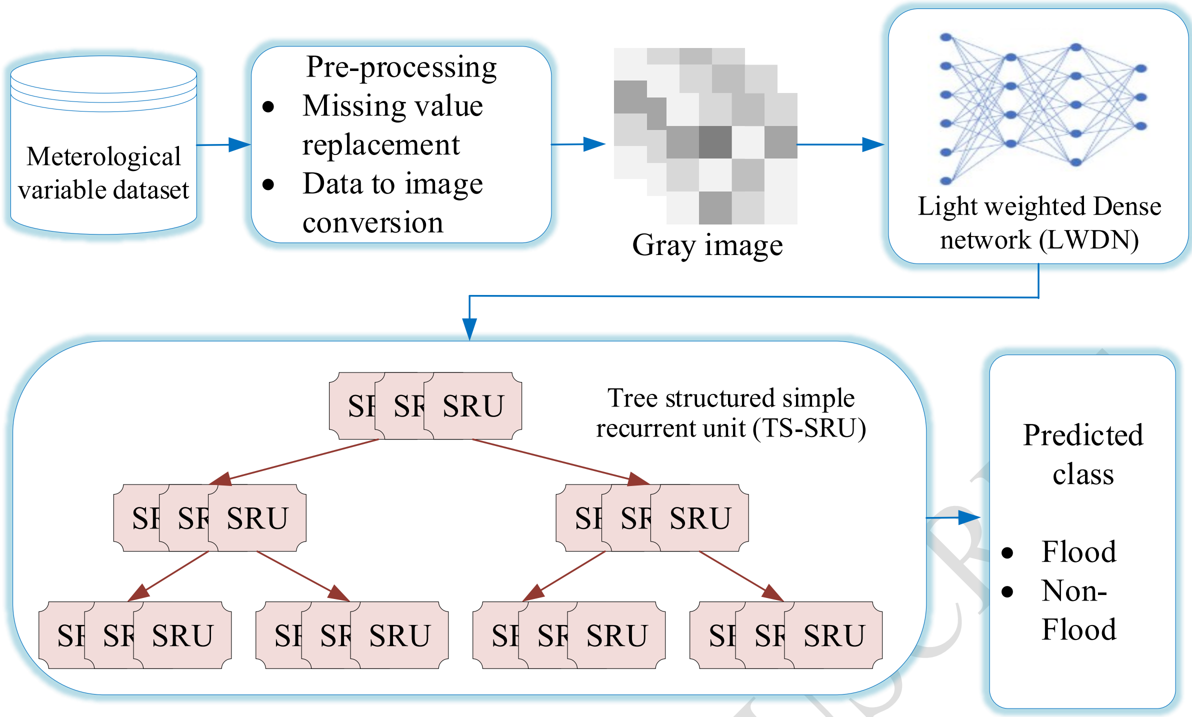


Figure 1. Overall structure of the proposed flood detection model

Figure 2 depicts the training procedure of the proposed TS-SRU. During the training process, the input dataset is split to get n distinct datasets for each classifier in the tree structure. These training sets are then utilized for training the classifiers in the respective nodes in the TS-SRU model. During the testing phase, the testing set is pre-processed before giving as input to trained TS-SRU model. Here, a Light Weighted Dense Network is added before the repeated SRU model to boost the characteristics of the input data.

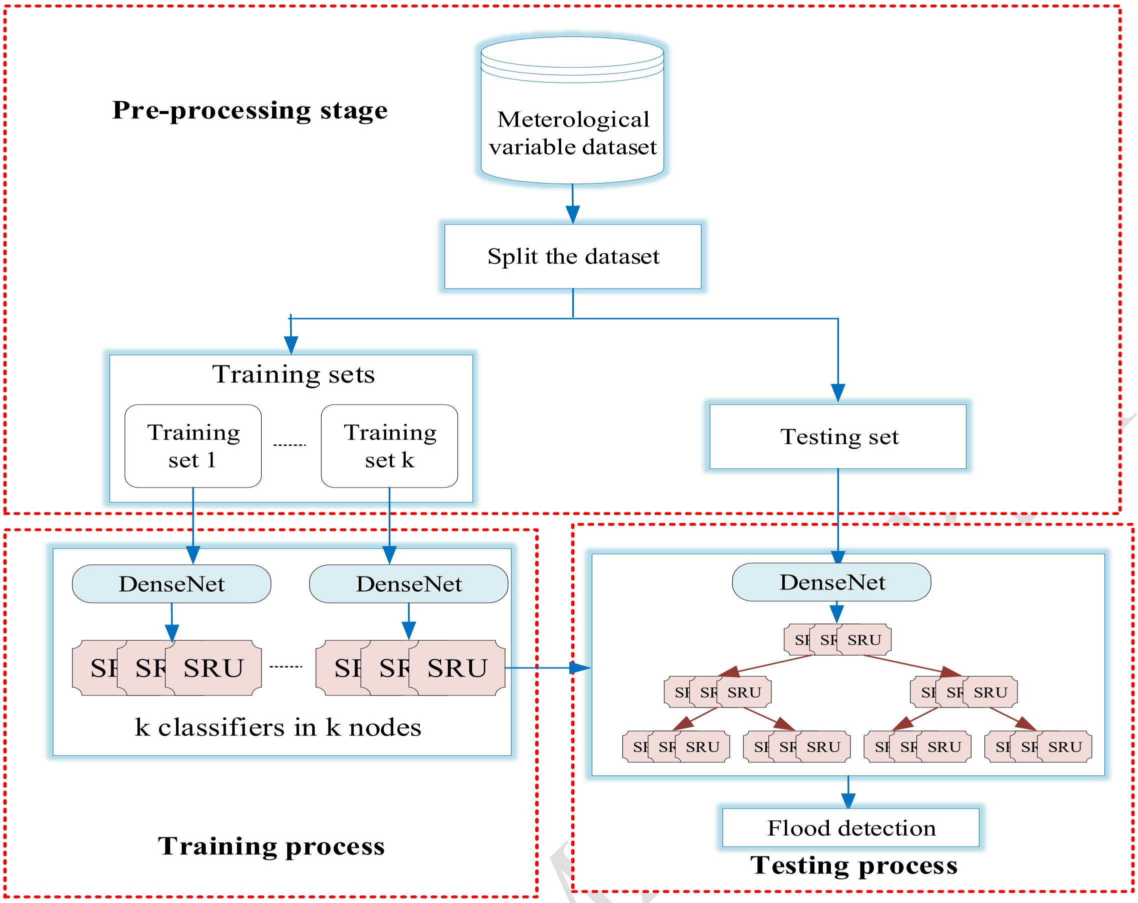


Figure 2. Training process of LDTSRU

3.1 Pre-processing

In this work, the neural network is trained using the data from the flood prediction dataset by Gauhar et al. (2021). The dataset includes data for 32 Bangladeshi districts. The dataset has several significant attributes, such as wind speed, minimum temperature, rainfall, cloud cover, and relative humidity. There may be anomalies (i.e. inaccurate or missing data) in the metrics acquired from different stations. This could be the result of problems with the sensors or even incorrect data storage. The suggested model pre-processes the data by substituting any random value from the same column for the missing values. After pre-processing the data, the input meteorological variable is arbitrarily divided into a training set and a testing set. Following that, a two-dimensional grayscale images can be obtained from the pre-processed data. In comparison to conventional methods, greyscale image representation of meteorological data offers a richer, more useful data representation by utilising the capabilities of deep learning and image processing to capture intricate spatial and inter-variable interactions. The reason behind this is that input data for the neural network model must have constant length. Different meteorological variables can be encoded into separate channels of an image. For instance, a 3-channel image could represent temperature, humidity, and wind speed. Analyzing these channels together allows models to capture how these variables interact spatially and temporally. After the conversion of meteorological variable into grayscale images, it can relate to image classification problem. Additional information about the pre-processing process is displayed in Algorithm 1.

Algorithm 1: Pre-processing
Input: Input meteorological variables; Classes M Output: Pre-processed dataset 1: for every j in $(1, M)$ do 2:   Divide data in each class M_j into M_{j1}, M_{j2},… M_{ji} 3: end for 4: Divide M into training and testing set 5: for every j in $(1, M)$ do 6:   for every i do 7:   obtain grayscale image from data 8:   end for 9: end for 10: Return Pre-processed dataset D

3.2. Light weighted Dense and Tree structural simple recurrent unit (LDTSRU)

The grayscale image obtained using the pre-processed meteorological data is assumed to have dimensions of $[\Re, \mathrm{C}]$, where $\Re$ rows and C columns make up the image. The time input is represented by the row and the number of features is represented by the column. Before the repeating SRU module, a light-weighted dense network is added to boost the input features. The output of the recurrent SRU module is sent to the output layer, where the anticipated outcomes are produced and the number of neurons equals the number of traffic classes.

3.2.1. Light weighted Dense network (LWDN)

In this work, the LWDN is introduced due to its simplicity since the prior features are only delivered to a subsequent layer through mathematical addition. It maintains a constant size between the input and the output while attenuating non-significant information and enhancing representative traits. The input and output feature maps are indicated as S_0 and S_6respectively in Figure 3. The feature maps $\{S_j\}_{j=1}^{5}$are attained using a sequence of convolution, back normalization, and ReLU operations.

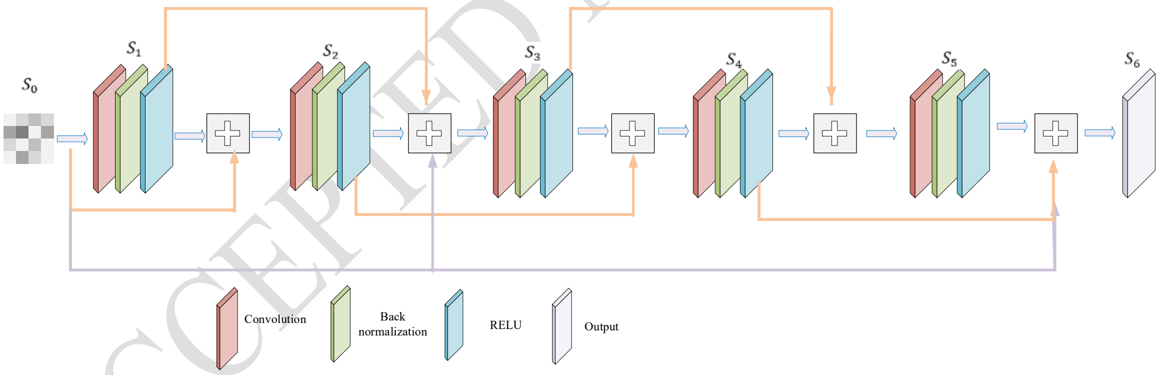


Figure 3. Light weighted Dense network (LWDN)

The following expressions are used to compute feature maps:

$$S_1 = seq_1^{2D}(S_0) \tag{1}$$

$$S_2 = seq_2^{2D}(S_0 + S_1) \tag{2}$$

$$S_3 = seq_3^{2D}(S_0 + S_1 + S_3) \tag{3}$$

$$S_4 = seq_4^{2D}(S_2 + S_3) \tag{4}$$

$$S_5 = seq_5^{2D}(S_3 + S_4) \tag{5}$$

$$S_6 = S_0 + S_4 + S_5 \tag{6}$$

where $\{seq_j^{2D}(.)|j = 1,2, \ldots ,5\}$ represent the sequence of operations of convolution, back normalization and ReLU in every stage. The feature transfer to deep layers is significantly encouraged by such a structure of connectedness.

3.2.2 Tree structured simple recurrent unit (TS-SRU)

Features extracted by the LWDN from an input image with $K = \Re \times \complement$ number of neurons are then input to the binomial tree layer. Here, the classification is improved through the usage of binomial tree structure, which refines the classification step by step. It makes use of the hierarchical structure of data for improving the performance and learning. Here, the node in the tree structure represents recurrent SRU as in Figure 4. The feature map is first obtained by the root node with recurrent SRU from the LWDN's final convolution layer. After that the number of nodes with recurrent SRU units increases one by one as the tree deepens. Here, the outcome of root node is divided into two vectors, each of which is sent to a different node for recurrent SRU in the following level. The output layer receives the output of the repeating SRU at the last level of the tree, where the number of neurons equals the number of flood classes and the anticipated outcomes are produced.

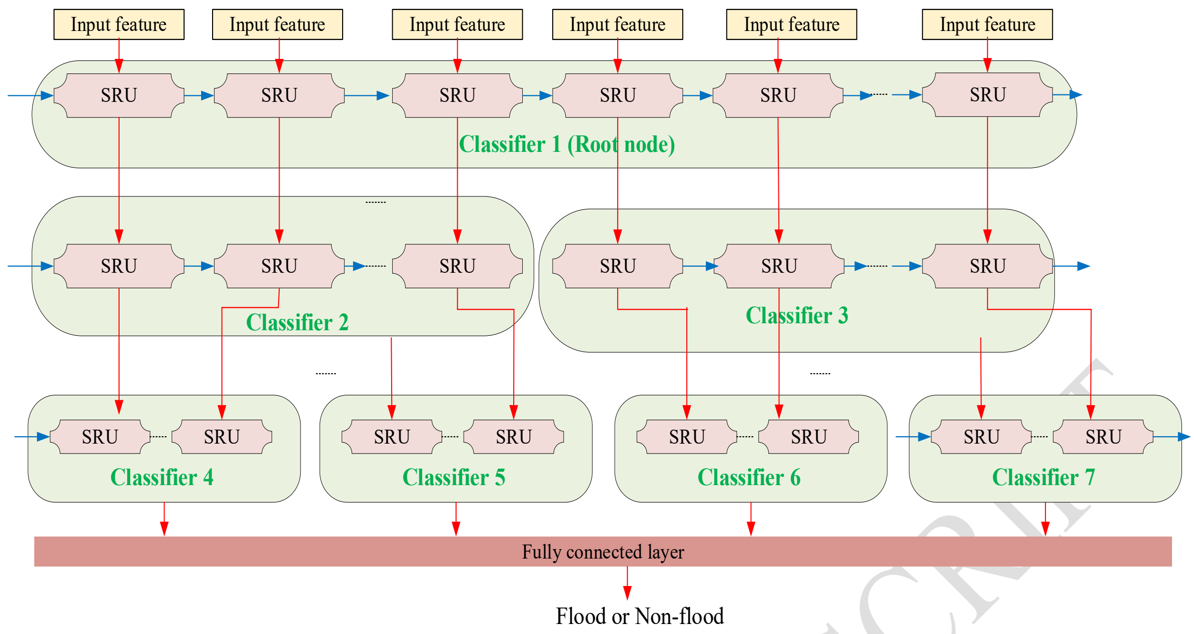


Figure 4. Tree structural simple recurrent unit

RNN is made up of loops where data is passed from one loop to the next. The chain-like characteristic suggests that RNN is suitable for sequences and lists due to its inherent structure. Let $\left[S_6^{(1)}, S_6^{(2)}, \dots, S_6^{(K)}\right]$ be the input sequence for RNN and its hidden state generates $[h_1, h_2, \dots h_K]$. The activation function $g$of the present input $S_6^{(n)}$and prior hidden state $h_{n-1}$is used to compute the hidden state at time-period n:

$$h_1 = g(S_6^{(n)}, h_{n-1}) \tag{7}$$

But the conventional RNN can't evade the issue of long-term dependences, which infers RNN would lose its ability to connect information as the distance between loops rises. As a result, unique varieties of RNN are suggested, including Gated Recurrent Units and Long Short-Term Memory. However, the calculation of each step depends on the previous one being finished. Therefore, parallelization is less appropriate for recurrent computations. An SRU has been designed to prevent this. In SRU, the gate calculation only dependent on the present input of recurrence. Thus, the only part that depends on earlier steps is the point-wise multiplication. As a result, parallelizing the matrix multiplications in the feed-forward network is simple.

Sequential dependency, memory limits, and backpropagation across time are the main obstacles to parallelising classical RNNs, which restricts their applicability to real-time flood prediction applications. Higher latency, less scalability, and greater resource requirements are the outcomes of these constraints. SRUs and CNNs, on the other hand, provide more parallel-friendly designs that make it possible to analyse big, complicated datasets more quickly and effectively. They are better suited for real-time flood prediction, when precise and timely forecasts are essential, due to their capacity to process geographical and temporal data simultaneously. The structure of SRU is shown in Figure 5.

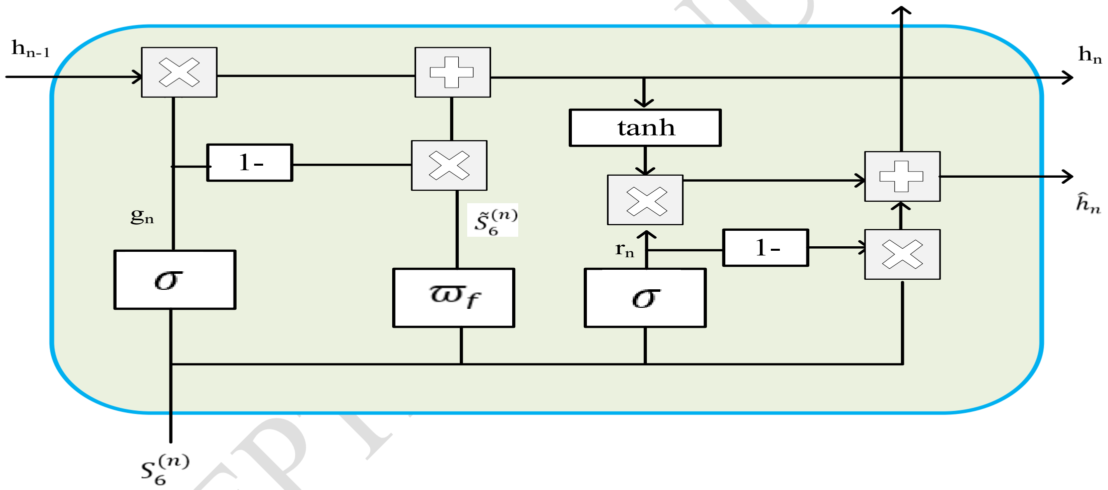


Figure 5. Simple recurrent unit

The SRU consist of a single forget gate. The linear alteration $\tilde{S}_6^{(n)}$and the forget gate $g_n$can be calculated using the input $S_6^{(n)}$at time $n$as follows:

$$\tilde{S}_6^{(n)} = \omega S_6^{(n)} \tag{8}$$

$$g_n = \sigma(\omega_g S_6^{(n)} + \beta_g) \tag{9}$$

The only need for this computation is $\tilde{S}_6^{(n)}$, which enables parallel processing over all time steps. The internal state$h_n$is modulated by the forget gate:

$$h_n = g_n \times h_{n-1} + (1 - g_n) \times \tilde{S}_6^{(n)} \tag{10}$$

The reset gate is used for computing the output state $\hat{h}_n$ by combining h_n and $S_6^{(n)}$ as given below:

$$r_n = \sigma(\omega_r S_6^{(n)} + \beta_n) \tag{11}$$

$$\hat{h}_n = r_n \times \tanh(h_n) + (1 - r_n) \times S_6^{(n)} \tag{12}$$

Algorithm 2 gives the training process of LDTSRU.

Algorithm 2: Training algorithm for Tree structural simple recurrent unit (TS-SRU)
Input: Training set of pre-processed datasets D_1^t, D_2^t, D_3^t, D_4^t, D_5^t; Maximum number of iteration M_{iter}; Batch size β Output: Files of model data Mdl_1, Mdl_2, Mdl_3 1: for every j in (1,2,…5) do 2: for every iteration in $(1, M_{iter})$ do 3: for every β samples in D_j^t do 4: Get feature map using LWDN 5: calculate loss function 6: Use Adam optimization algorithm 7: Update the weights and biases 8: end for 9: end for 10: Generate files of model data Mdl_i 11: end for

4. Results and discussion

In this section, the performance of the proposed model has been validated through simulation. The Python programming language is used to simulate the suggested flood detection approach. An open-source deep learning framework, TensorFlow served as the foundation for the design of the suggested LDTSRU model. It had Intel (R) Core (TM) i5-9300h CPU (main frequency = 2.4 GHz), a Win10 64-bit operating system, and an NVIDIA GTX1660ti video card. The host was configured with CUDA 10.0 and cudnn7.3 in order to speed up the card's functioning. The Adam optimization approach with a mini-batch size of 50 was utilized to train the proposed model across 200 epochs. The first momentum decay rate was 0.90, and the second was 0.99. The initial learning rate was 0.001.

The performance of the suggested model is verified using 65 years of weather data from Bangladesh and is available in https://www.kaggle.com/datasets/emonreza/65-years-of-weather-data-bangladesh-preprocessed. The Bangladesh Meteorological Department (BMD) provided the data. The data on the frequency of flooding for a given month and year was gathered from multiple sources, such as yearly flood reports, newspapers, research papers, etc., and combined with BMD's weather data to produce an updated dataset with 20544 occurrences that can be found in Chithra et al. (2022). The dataset includes data for 32 Bangladeshi districts. The dataset has several significant attributes, such as wind speed, minimum temperature, rainfall, cloud cover, and relative humidity. The dissimilar features could negatively impact the model and produce undesirable outcomes; thus, they should not be selected. This section first examines the correlation between each meteorological variable employed for flood detection as in Figure 6. The statistical relationship between any two variables is called correlation. This analysis shows that the correlation between the selected features in the dataset is good. Hence, they can provide good performance for flood forecasting.

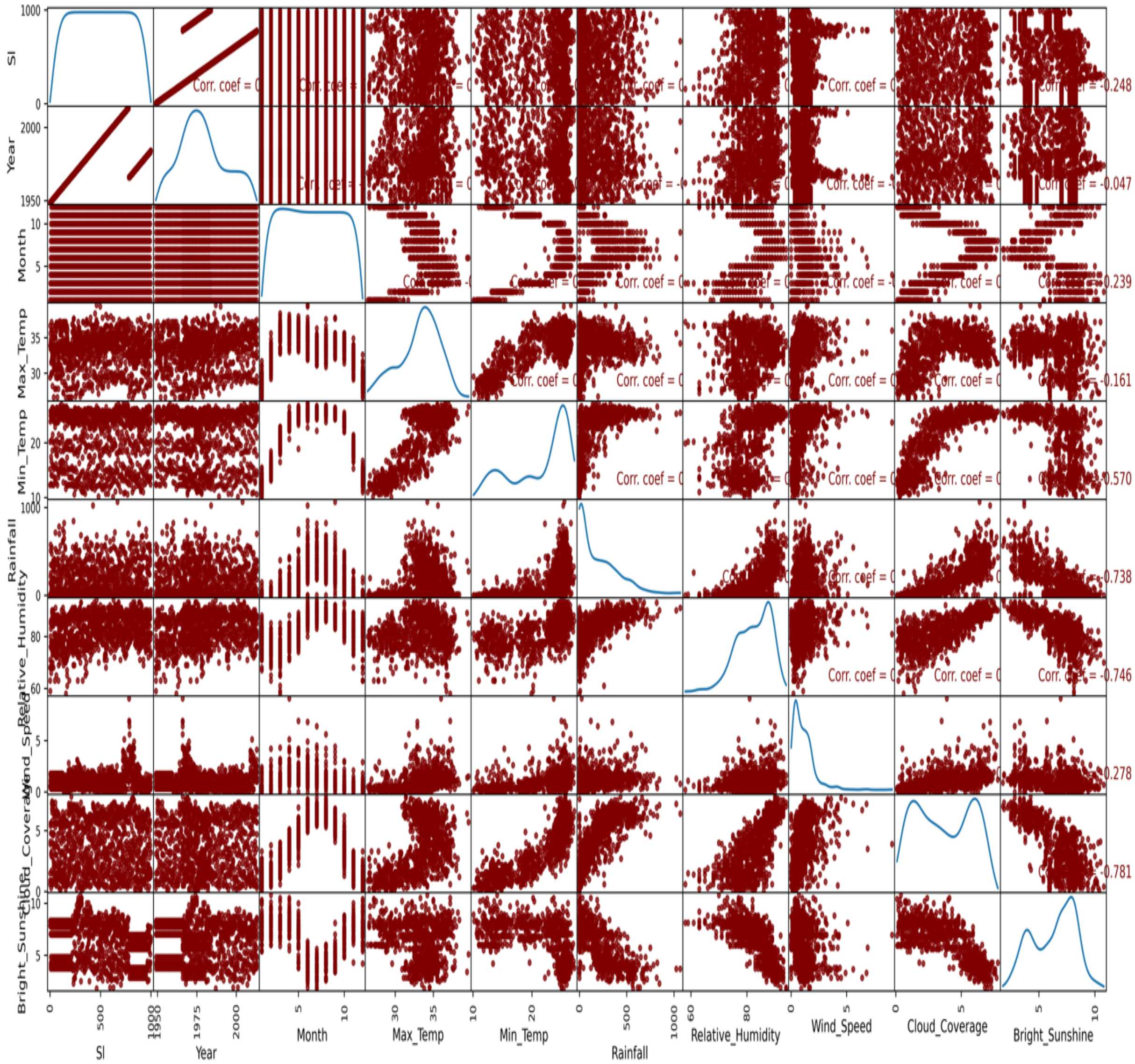


Figure 6. Correlation analysis between each meteorological variable

4.1. Performance metrics

Three assessment metrics are used to evaluate the prediction performance of the suggested model. The accuracy of flood detection is measured by the Accuracy (A) metric.

$$A = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \tag{13}$$

The fraction of false negatives that are found is known as specificity (S). The following formula determines specificity:

$$S = \frac{T_P}{T_N + F_P} \tag{14}$$

One metric for assessing how many of the features of the solution that are accurate based on data is precision (P):

$$P = \frac{T_P}{T_P + F_P} \tag{15}$$

The number of data features that the suggested method precisely recovers is determined by recall (R).

$$R = \frac{T_P}{T_P + F_P} \tag{16}$$

The mean values of recall and precision are used to compute the F-measure(F), which can be formally defined as follows:

$$F = 2 \times \frac{P \times R}{P + R} \tag{17}$$

4.2. Evaluation of the proposed model

Initially, the perfromance of the proposed model is validated by comparing it with some of the base line models including CNN, RNN, LSTM, GRU, and SRU. The performance of the suggested model is compared in Table 1 with respect to F-measure, Accuracy, Specificity, Precision, and Recall. While CNNs are very good at capturing spatial patterns, they are not very good at modeling temporal dependencies, which are important when trying to understand how meteorological factors change over time in order to forecast floods. Because of its hierarchical structure, parallel processing capabilities, and effectiveness in managing lengthy sequences, the TS-SRU provides a number of advantages when it comes to managing temporal dependencies and event flow. These qualities make it an attractive option for applications such as the analysis of meteorological data, where the ability to capture intricate, multi-scale connections is essential. Hierarchical structures are used in many real-world tasks, including several forms of meteorological data analysis. Since TS-SRU can naturally describe the data hierarchy, it is a good fit for these kinds of activities.

Furthermore, typical RNNs are not very good at remembering information from far back in the sequence. But flood prediction frequently necessitates long-term dependencies. LSTMs and GRUs are effective tools to performing complex sequence modeling problems, but they have significant drawbacks as compared to Simple Recurrent Units (SRUs). In some circumstances, LSTMs and GRUs are less efficient because of their higher computational complexity, longer training periods, increased model complexity, larger memory requirements, and slower inference rates. Nonetheless, the suggested LDTSRU improves the efficacy and precision of flood detection models by utilizing the advantages of SRU and CNN architectures. Additionally, the SRU tree-structured model makes it possible to comprehend the order and flow of events that culminate in a flood effectively. Thus, the proposed model performance is much higher than the existing single model.

Table 1. Comparison with baseline models

Model	*A*	*S*	*P*	*R*	*F*
CNN	80.37	81.34	79.16	81.72	82.35
RNN	83.16	84.34	83.94	82.67	84.38
LSTM	87.88	86.23	85.74	83.82	85.63
SRU	88.01	86.43	86.17	84.24	85.97
LDTSRU	98.68	97.13	97.67	97.79	96.34

Figure 7 displays the differences in accuracy between the models on the training set during the training procedure. It shows that after 200 iterations, LDTSRU has an accuracy of more than 90%. Additionally, the area under the ROC curve (AUC) was computed in order to compare the results with other baseline models developed by Periasamy et al (2024), Santhanaraj et al. (2023), Surendran et al. (2023) and Surendran et al. (2021). The model's capacity to discriminate between flood and non-flood occurrences is evaluated with the aid of ROC analysis. The AUCs of the LDTSRU are higher than those of the other models, as seen in Figure 7 (b) and a value closer to 1 denotes a better-performing model.

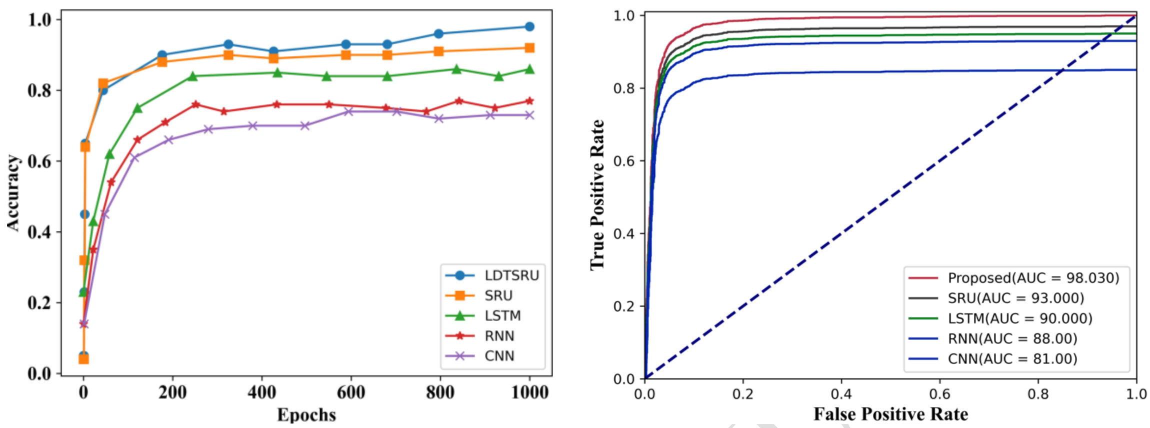


Figure 7. Performance analysis (a) Accuracy plot (b) ROC plot

The number of hidden units and step size were changed in order to verify the model performance and training speed of LDTSRU. The resulting data are shown in Figure 8. The training time of the model is significantly impacted by increases in the number of hidden layer units and training step size. The training time of every model has been raised as the step size grew. The LSTM model required more time to train. The GRU and SRU models had the longest and shortest training times, respectively. Furthermore, when the number of hidden units, the alteration in training time became more noticeable. The training time of LSTM and GRU have been increased while increasing the number of hidden units because LSTM and GRU has serial input structures. Also, Additionally, the curve's slope and amplitude were both quite big. But the curve amplitude and slope were quite minimal for SRU due to its distinct parallel structure. There was no discernible difference between the SRU and LSTM prediction effects when the identical input dataset was used. But the proposed LDTSRU model integrates SRU with Light weighted Dense network to improve the performance compared to SRU. But the training time of the proposed model is slightly higher than the SRU model. It can be negotiated because, the accuracy of LDTSRU is much better than the SRU model.

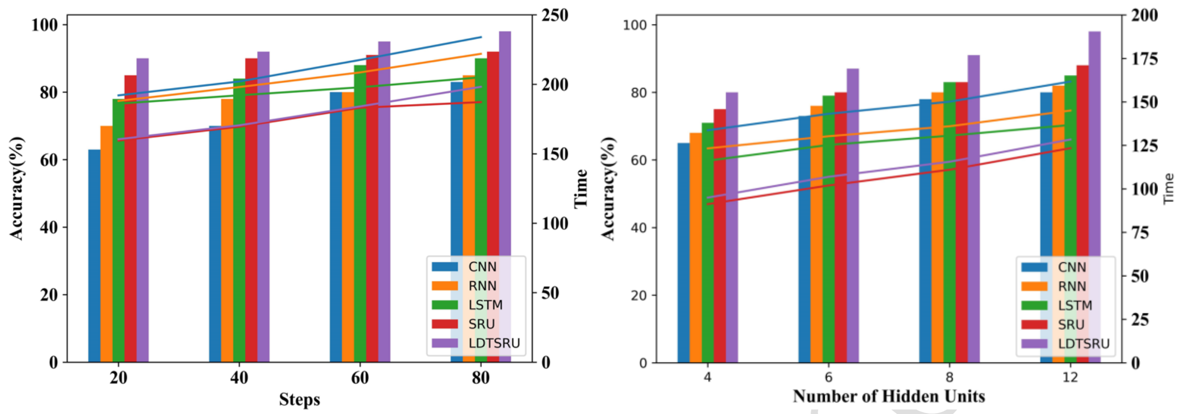


Figure 8. Analysis by adjusting (a) step size (b) Hidden units

The performance of the suggested draught prediction is compared with state-of-the-art methods including LSTM, Explainable Artificial Intelligence (EAI) and IPCA-CNN in Table 2. Here, the KNN used the same weather data of Bangladesh for 65 years. However, the remaining models were built using a dataset that included daily and monthly rainfall data for Kerala state, acquired between 1901 and 2021. Based on 70% training and 30% test datasets, all of these approaches display the relative performance of the flood forecasting outputs. This investigation demonstrates that the existing LSTM has the lowest performance when compared to the current IPCA-CNN, EAI, and the recommended model. But the proposed LDTSRU classifier obtains 98.68% accuracy, whereas the existent IPCA-CNN attains 96.24% accuracy, EAI achieved 92.90% accuracy and LSTM had 88.79%. This is due to the integration of depth network, SRU and tree structure in the proposed model. The lightweight depth network efficiently increases the features of the input data. The binomial tree structured SRU is used to improve the classification by successively refining the classification.

Table 2. Comparison with other reported works

Model	*A*	*S*	*P*	*R*	*F*
KNN	94.91	-	92.50	91.00	92.00
LSTM	88.79	85.34	84.83	84.74	84.74
EAI	92.90	89.31	86.47	86.19	86.19
IPCA-CNN	96.24	93.48	94.23	93.29	93.29
LDTSRU	98.68	97.13	97.67	97.79	96.34

Adapting LDTSRU with real-time data sources, optimising for scalability and efficiency, and putting in place reliable monitoring and maintenance routines are all necessary for adapting it for real-time flood monitoring and early warning systems. Assuring data quality, cutting down on latency, preserving model robustness, and overseeing operational integration are important difficulties. Transforming a research prototype into a workable, scalable solution requires careful planning, testing, and continual development to address these issues. A comprehensive approach is needed to ensure the robustness and reliability of LDTSRU predictions in a range of environmental and climatic conditions. This approach should include diverse training data, resilient data processing pipelines, adaptive learning mechanisms, robust model architectures, and rigorous evaluation. All of these steps work together to improve the model's high prediction accuracy and generalisation across various situations, which makes it a workable option for early warning systems and real-time flood monitoring. By addressing long-range dependencies, pooling predictions from several models to improve robustness and reduce mistakes, and improving emphasis on relevant characteristics, the integration of attention mechanisms and ensemble learning approaches into LDTSRU can greatly improve its predictive powers. While ensemble learning increases overall performance and adaptation to changing data circumstances, attention processes provide interpretability and contextual awareness. Effective implementation in operational flood monitoring and early warning systems requires striking a balance between these improvements and computational efficiency as well as real-time constraints.

5. Conclusion

The proposed LDTSRU offers excellent prospects for accurate and timely decision-making for disaster management systems. It mainly aims to forecast floods by combining meteorological data's spatial and temporal properties. To capture intricate, non-linear interactions between features and the target variable, this model uses a specific type of binary tree. The classifier learns the time-related properties of the data using the SRU model. The LDTSRU model trained more quickly than previous models while maintaining prediction accuracy. It has greatly improved the operating efficiency and reduced the training time abruptly. According to experimental data, the LDTSRU approach has some gains in flood prediction accuracy (98.68%) over previous methods, which offers the metrological department a smart and affordable idea. In the future, the proposed model could be improved by including optimization method to help it learn more quickly and analyse data more efficiently without reducing the number of attributes in real-time.

Data Availability Statement:

The source of data sets are download from the following link

"Flood-prediction" github.com. https://github.com/ngauhar/Flood-prediction

References

Saravanan, Subbarayan, Devanantham Abijith, Nagireddy Masthan Reddy, K. S. S. Parthasarathy, Niraimathi Janardhanam, Subbarayan Sathiyamurthi, and Vivek

Sivakumar. "Flood susceptibility mapping using machine learning boosting algorithms techniques in Idukki district of Kerala India." Urban Climate 49 (2023): 101503.

Munawar, Hafiz Suliman, Ahmed WA Hammad, and S. Travis Waller. "Remote sensing methods for flood prediction: A review." Sensors 22, no. 3 (2022): 960.

Surendran R., Tamilvizhi T., and Lakshmi, S. (2021), Integrating the Meteorological Data into a Smart City Service Using Cloud of Things (CoT). In Emerging Technologies in Computing: 4th EAI/IAER International Conference, iCETiC 2021, Virtual Event, August 18–19, 2021, Springer International Publishing, 4, 94-111.

Panahi, Mahdi, Abolfazl Jaafari, Ataollah Shirzadi, Himan Shahabi, Omid Rahmati, Ebrahim Omidvar, Saro Lee, and Dieu Tien Bui. "Deep learning neural networks for spatially explicit prediction of flash flood probability." Geoscience Frontiers 12, no. 3 (2021): 101076.

Wang, Yakun, Zhimin Fu, Zhaoyang Cheng, Yan Xiang, Jiankang Chen, Pengtao Zhang, and Xin Yang. "Uncertainty analysis of dam-break flood risk consequences under the influence of non-structural measures." International Journal of Disaster Risk Reduction 102 (2024): 104265.

Jin, Hengxu, Haipeng Lu, Yu Zhao, Zhizhou Zhu, Wujie Yan, Qiqi Yang, and Shuliang Zhang. "Integration of an improved transformer with physical models for the spatiotemporal simulation of urban flooding depths." Journal of Hydrology: Regional Studies 51 (2024): 101627.

Shivarudrappa, Halappanavar Ruta, S. P. Nandhini, T. S. Pushpa, and K. P. Shailaja. "Enhancing Flood Forecasting Accuracy Through Machine Learning Approaches." In International Conference on Interdisciplinary Approaches in Civil Engineering for Sustainable Development, pp. 203-211. Singapore: Springer Nature Singapore, 2023.

Chen, Chen, Jiange Jiang, Zhan Liao, Yang Zhou, Hao Wang, and Qingqi Pei. "A short-term flood prediction based on spatial deep learning network: A case study for Xi County, China." Journal of Hydrology 607 (2022): 127535.

Yu, Qiying, Chen Shi, Yungang Bai, Jianghui Zhang, Zhenlin Lu, Yingying Xu, Wenzhong Li, Chengshuai Liu, Lu Tian, and Caihong Hu. "Interpretable baseflow segmentation and prediction based on numerical experiments and deep learning." Journal of Environmental Management 360 (2024): 121089.

Gurbuz, Faruk, Avinash Mudireddy, Ricardo Mantilla, and Shaoping Xiao. "Using a physics-based hydrological model and storm transposition to investigate machine-learning algorithms for streamflow prediction." Journal of Hydrology 628 (2024): 130504.

Aljohani, Fares Hamad, Ahmad B. Alkhodre, Adnan Ahamad Abi Sen, Muhammad Sher Ramazan, Bandar Alzahrani, and Muhammad Shoaib Siddiqui. "Flood Prediction using Hydrologic and ML-based Modeling: A Systematic Review." International Journal of Advanced Computer Science & Applications 14, no. 11 (2023).

Rasheed, Zimeena, Akshay Aravamudan, Ali Gorji Sefidmazgi, Georgios C. Anagnostopoulos, and Efthymios I. Nikolopoulos. "Advancing flood warning procedures in ungauged basins with machine learning." Journal of Hydrology 609 (2022): 127736.

Hasan, Md Hasibul, Asib Ahmed, K. M. Nafee, and Md Amzed Hossen. "Use of machine learning algorithms to assess flood susceptibility in the coastal area of Bangladesh." Ocean & Coastal Management 236 (2023): 106503.

Costache, Romulus, Alireza Arabameri, Iulia Costache, Anca Crăciun, Abu Reza Md Towfiqul Islam, Sani Isah Abba, Mehebub Sahana, Manish Pandey, Tran Trung Tin, and Binh Thai Pham. "Flood hazard potential evaluation using decision tree state-of-the-art models." Risk Analysis 44, no. 2 (2024): 439-458.

Mehravar, Soroosh, Seyed Vahid Razavi-Termeh, Armin Moghimi, Babak Ranjgar, Fatemeh Foroughnia, and Meisam Amani. "Flood susceptibility mapping using multi-temporal SAR imagery and novel integration of nature-inspired algorithms into support vector regression." Journal of Hydrology 617 (2023): 129100.

Aiyelokun, O. O., O. D. Aiyelokun, and O. A. Agbede. "Application of random forest (RF) for flood levels prediction in Lower Ogun Basin, Nigeria." Natural Hazards 119, no. 3 (2023): 2179-2195.

Fang, Zhice, Yi Wang, Ling Peng, and Haoyuan Hong. "Predicting flood susceptibility using LSTM neural networks." Journal of Hydrology 594 (2021): 125734.

Luppichini, Marco, Michele Barsanti, Roberto Giannecchini, and Monica Bini. "Deep learning models to predict flood events in fast-flowing watersheds." Science of The Total Environment 813 (2022): 151885.

Wei, Chen, Lan-lan Chen, Zhen-zhen Song, Xiao-guang Lou, and Dong-dong Li. "EEG-based emotion recognition using simple recurrent units network and ensemble learning." Biomedical signal processing and control 58 (2020): 101756.

Ling, Jie, Zhishen Zhu, Yu Luo, and Hao Wang. "An intrusion detection method for industrial control systems based on bidirectional simple recurrent unit." Computers & Electrical Engineering 91 (2021): 107049.

Khan, Izhar Ahmed, Nour Moustafa, Imran Razzak, Muhammad Tanveer, Dechang Pi, Yue Pan, and Bakht Sher Ali. "XSRU-IoMT: Explainable simple recurrent units for threat detection in Internet of Medical Things networks." Future generation computer systems 127 (2022): 181-193.

Surendran R., Alotaibi Y. and Subahi A.F. (2023), Wind Speed Prediction Using Chicken Swarm Optimization with Deep Learning Model. Computer Systems Science & Engineering, 46, 3.

Kimura, Nobuaki, Ikuo Yoshinaga, Kenji Sekijima, Issaku Azechi, and Daichi Baba. "Convolutional neural network coupled with a transfer-learning approach for time-series flood predictions." Water 12, no. 1 (2019): 96.

Chen, Chen, Qiang Hui, Wenxuan Xie, Shaohua Wan, Yang Zhou, and Qingqi Pei. "Convolutional Neural Networks for forecasting flood process in Internet-of-Things enabled smart city." Computer Networks 186 (2021): 107744.

Sawaf, Mohamad Basel Al, Kiyosi Kawanisi, Mohamad Nazieh Jlilati, Cong Xiao, and Masoud Bahreinimotlagh. "Extent of detection of hidden relationships among different hydrological variables during floods using data-driven models." Environmental Monitoring and Assessment 193 (2021): 1-14.

Balamurugan, R., Kshitiz Choudhary, and S. P. Raja. "Prediction of flooding due to heavy rainfall in India using machine learning algorithms: providing advanced warning." IEEE Systems, Man, and Cybernetics Magazine 8, no. 4 (2022): 26-33.

Kaur, Mandeep, Pankaj Deep Kaur, and Sandeep K. Sood. "ANFIS-based flood detection and vulnerability assessment framework." Hydrological Sciences Journal 67, no. 15 (2022): 2310-2326.

Surendran, R., Alotaibi, Y. and Subahi, A.F., 2023. Lens-Oppositional Wild Geese Optimization Based Clustering Scheme for Wireless Sensor Networks Assists Real Time Disaster Management. Comput. Syst. Sci. Eng., 46(1), pp.835-851.

Singh, Dharmaveer, Kunal Karan, Sudhir Kumar Singh, Pankaj Chauhan, and Ronny Berndtsson. "Fusing morphometric characteristics with extreme precipitation indices for identifying the most vulnerable sub-basin at risk of flooding." Journal of Hydro-environment Research 50 (2023): 44-56.

Du, Jinyang, John S. Kimball, Justin Sheffield, Ming Pan, Colby K. Fisher, Hylke E. Beck, and Eric F. Wood. "Satellite flood inundation assessment and forecast using SMAP and landsat." IEEE journal of selected topics in applied earth observations and remote sensing 14 (2021): 6707-6715.

Gharakhanlou, Navid Mahdizadeh, and Liliana Perez. "Flood susceptible prediction through the use of geospatial variables and machine learning methods." Journal of hydrology 617 (2023): 129121.

Letessier, Clara, Jean Cardi, Antony Dussel, Isa Ebtehaj, and Hossein Bonakdari. "Enhancing flood prediction accuracy through integration of meteorological parameters in river flow observations: A case study Ottawa river." Hydrology 10, no. 8 (2023): 164.

John, Tegil J., and R. Nagaraj. "Prediction of floods using improved pca with one-dimensional convolutional neural network." International Journal of Intelligent Networks 4 (2023): 122-129.

Santhanaraj R. K., Rajendran S., Romero C. A. T., and Murugaraj, S. S. (2023), Internet of Things Enabled Energy Aware Metaheuristic Clustering for Real Time Disaster Management. Comput. Syst. Sci. Eng., 45, 1561-1576.

Gauhar, Noushin, Sunanda Das, and Khadiza Sarwar Moury. "Prediction of flood in Bangladesh using K-nearest neighbors algorithm." In 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), pp. 357-361. IEEE, 2021.

Chithra, K., B. V. Binoy, and P. Bimal. "Spatial mapping of the flood-affected regions of Northern Kerala: a case study of 2018 Kerala floods." Journal of the Indian Society of Remote Sensing 50, no. 4 (2022): 677-688.

Periasamy, S., Subramanian, P. and Surendran, R., 2024. An intelligent air quality monitoring system using quality indicators and transfer learning based lightweight recurrent network with skip connection.

Chen, G., Zhang, K., Wang, S., Xia, Y., & Chao, L. (2023). iHydroSlide3D v1.0: an advanced hydrological–geotechnical model for hydrological simulation and three-dimensional landslide prediction. Geosci. Model Dev., 16(10), 2915-2937. doi: https://doi.org/10.5194/gmd-16-2915-2023

Zhao, Y., Li, J., Wang, Y., Zhang, W., & Wen, D. (2024). Warming Climate-Induced Changes in Cloud Vertical Distribution Possibly Exacerbate Intra-Atmospheric Heating Over the Tibetan Plateau. Geophysical Research Letters, 51(3), e2023GL107713. doi: https://doi.org/10.1029/2023GL107713

Li, J., Zhao, Y., Chen, D., Zhao, P., Zhang, C.,... Wang, Y. (2024). The Quantitative Role of Moisture and Vertical Motion in Shaping Summer Heavy Rainfall over North China under Two Distinct Large-Scale Weather Patterns. Journal of Climate, 37(8), 2655-2672. doi: https://doi.org/10.1175/JCLI-D-22-0850.1

Li, R., Zhu, G., Lu, S., Sang, L., Meng, G., Chen, L.,... Wang, Q. (2023). Effects of urbanization on the water cycle in the Shiyang River basin: based on a stable isotope method. Hydrology and Earth System Sciences, 27(24), 4437-4452. doi: 10.5194/hess-27-4437-2023

Xie, X., Xie, B., Cheng, J., Chu, Q., & Dooling, T. (2021). A simple Monte Carlo method for estimating the chance of a cyclone impact. Natural Hazards, 107(3), 2573-2582. doi: 10.1007/s11069-021-04505-2

Wu, L., He, B., & Peng, J. (2024). Analysis of Rainfall-Caused Seepage into Underlying Bedrock Slope Based on Seepage Deformation Coupling. International Journal of Geomechanics, 24(5), 04024076. doi: https://doi.org/10.1061/IJGNAI.GMENG-9175