Comparative study of Models used to predict the overflow probability of various schemes of Narmada Basin

Aryan Raje[1]
Department of Computer Engineering
Vivekanand Education Society's
Institute Of Technology (Affiliated to the University of Mumbai)
Mumbai, India

Arya Raje[2]
Department of Computer Engineering
Vivekanand Education Society's
Institute Of Technology (Affiliated to the University of Mumbai)
Mumbai, India

Ishita Marathe[3]
Department of Computer Engineering
Vivekanand Education Society's
Institute Of Technology (Affiliated to the University of Mumbai)
Mumbai, India

Prasad Lahane[4]
Department of Computer Engineering
Vivekanand Education Society's
Institute Of Technology (Affiliated to the University of Mumbai)
Mumbai, India

Mrs. Gresha Bhatia[5]
(Deputy H.O.D)
Department of Computer Engineering
Vivekanand Education Society's
Institute Of Technology (Affiliated to the University of Mumbai)
Mumbai, India

***Abstract*—This paper conducts a comparative study on various models predicting overflow probability in different schemes within the Narmada Basin, a crucial water resource in central India. By employing statistical, hydrological, and machine learning methods and considering factors like rainfall patterns and basin topography, the study assesses model performance. Through literature review and empirical data validation, it aims to identify strengths and weaknesses in capturing overflow dynamics. The insights gained contribute to enhancing flood forecasting and management strategies, supporting sustainable water resource utilization in the region.**

Keywords—Deep Learning, Comparative Study, Narmada Basin,

I. INTRODUCTION

The release of approximately 18 lakh cusecs of water from the Sardar Sarovar dam on September 17, 2023, resulted in floods in Narmada, Bharuch, and parts of Vadodara district for two days. SSNNL reported an inflow of nearly 22 lakh cusecs, managed through diverting some water into canals for filling water bodies in other areas. The incident, attributed to state government mismanagement, highlights the need for a predictive monitoring system to prevent sudden water overflow. Our aim is to develop a system which can avoid such mishaps from happening again

Many mathematical and mathematical models have been developed to predict floods, and many model comparisons and studies have been conducted using data models such as machine learning [3].Various machine learning algorithms are available, each tailored to different output requirements and datasets. For flood prediction, supervised learning models are deemed most effective [3].

II. RELATED WORK

In [1], authors utilize a random forest model to identify high-risk flood areas.

In [2], authors use advanced models in GIS to produce flood-susceptibility maps for Seoul, highlighting key factors like proximity to rivers and topography, with validation accuracy.

In [3], authors have presented an overview of numerous machine learning algorithms which are used in flood predictions. These algorithms are compared and the one which best suits is selected for the model.

In [4], authors have tackled flood issues in the lower Narmada basin, using flood frequency analysis and hydrodynamic simulation to predict peak floods and produce inundation maps. It determines Log-Pearson Type III Distribution for lower return periods and Gumbel's method for higher ones, providing essential guidance for flood mitigation planning.

In [5], forecasting systems by the Central Water Commission rely on statistical methods and real-time data collection from field stations via wireless, telephone/mobile, and satellite telemetry. In-house developed models utilize rainfall data from various sources for predictions, disseminated through dedicated websites and social media.

In [6], The Ministry initiated the "Flood Management Programme (FMP)" as a State sector Scheme during the XI Plan, and it persisted through the XII Plan. A total of 522

projects, amounting to Rs. 13238.37 crore, were approved and incorporated within the FMP.

In [7], authors have enhanced flood forecasting at Sardar Sarovar Dam by identifying crucial rain gauge stations using clustering and Thiessen polygons.. It concludes that the existing gauge network suffices for forecasting and stresses the importance of selecting appropriate stations for accurate predictions in flood-prone areas

III. TRADITIONAL AND EXISTING SYSTEM

Existing systems have lack of consideration for dynamic factors such as changing land use patterns and infrastructure development [1].They also include potential inaccuracies due to assumptions in model inputs and uncertainties in future climate change projections [2]. Inadequate or less number of parameters also drops the accuracy of the model [3]. The study might not be entirely accurate because there might not be enough or reliable historical flood data. Flood maps created through simulation could have uncertainties because they rely on certain guesses and simplifications [4].In existing models used by government the nodal officer of the dam/reservoir share reservoir related data with CWC through uploading on Water Information Management System (WIMS) or sending through email/SMS/Phone/Wireless etc and have to wait for CWC to process the data and decide course of action this time consuming chain of action can prove fatal in need of emergency [5]. Majority of the project are heavily inclined towards flood management and hydrological studies, and do not stress any importance on prediction as an effective measure [6]. Short data period poses potential challenges in generalizing the model beyond specific years, and oversimplified assumptions about rainfall-runoff relationships and land characteristics [7]

IV. RANDOM FOREST REGRESSION FOR FLOOD PREDICTION

Random Forest works as a classifier by utilizing multiple decision trees trained on diverse subsets of the dataset. Through aggregating the predictions of these trees via majority voting, it enhances overall predictive accuracy, diverging from the reliance on a single decision tree. Increasing the number of trees in the forest not only boosts accuracy but also addresses overfitting concerns.

It's tailored to integrate real-time data on water levels and hydrological factors from various schemes of the Narmada river. The primary goal is to bolster disaster preparedness and mitigate flood risks effectively.

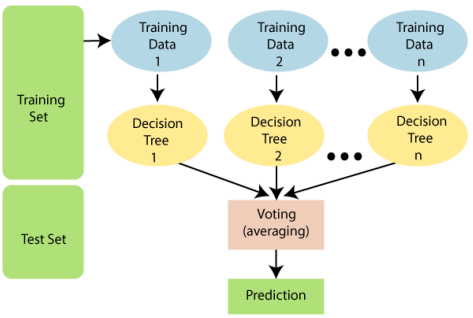

Fig. 1 Working of Random Forest Regression

V. PROPOSED SYSTEM

Machine learning algorithms employ an automated inductive method to extract insights from data and identify patterns, facilitating the creation of prediction models. These models are then preprocessed and new datasets generate predictions based on the accuracy and precision of the models. The use of machine learning is essential because it enables the processing of large volumes of data, which can be inputted into the algorithms and trained using supervised or unsupervised learning techniques [8]. This process aims to categorize the data gathered from the compiled dataset.

In the proposed model, the dataset is collected from https://wrd.guj.nic.in/dam/hour_reports_h.php and the various input data like Design gross Storage, Rule Level, Present Gross Storage, Outflow River, Cum.Rainfall, gate-Position-Nos, Scheme, FRL, Present-Water Level(m), Inflow, Outflow Canal, Type of Gate, Opening, Using this data, a relationship between the level of the reservoir and the gate opening of the reservoir can be established which can be used to train the random Forest Regression Model.

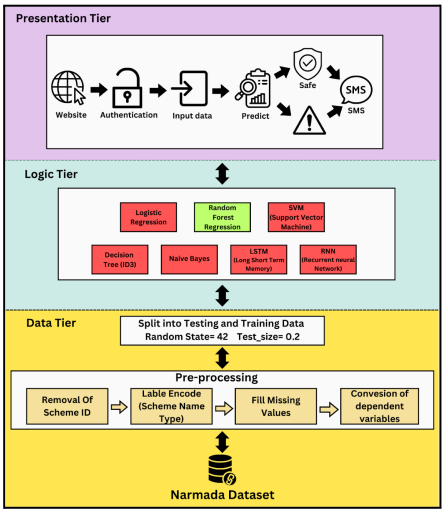

Fig. 2 Flow of Proposed System

VI. ALGORITHM AND PROCESS DESIGN

1. Data Preprocessing

Loading Dataset: Initially, the dataset containing information about various water reservoirs, including scheme names, design parameters, and historical storage levels, is uploaded using the Pandas library.

Data Cleaning: Irrelevant features are removed, and missing values are filled with the mean of respective columns to ensure a complete dataset.

Encoding Categorical Variables: Categorical variables like "Scheme Name" and "Type" are encoded into numerical format using label encoding for compatibility with machine learning models.

2. Model Training

Splitting Data: The dataset undergoes division into training and testing sets employing the train_test_split function available in Scikit-learn. This process guarantees the assessment of the model's performance on data that it has not previously encountered.

Random Forest Classifier: A Random Forest Classifier is called a predictive model due to its vigor and capability to maneuver both numerical data and categorical data effectively.

3. Graphical User Interface (GUI) Development

Creating Tkinter Window: A GUI is developed using the Flask , providing an intuitive interface for users to input reservoir parameters and obtain predictions.

Input Fields and Labels: Input fields and labels are dynamically generated within the GUI, allowing users to input relevant parameters such as scheme name, design parameters, and environmental factors.

4. Prediction Execution

Prediction Functionality: Upon user input, the application triggers a function to predict the percentage storage of the specified reservoir based on the trained Random Forest model.

Result Display: Predicted results are displayed to the user through a message box, indicating whether the reservoir's storage level is within safe limits or if there's a risk of overflow.

5. Model Persistence

Model Saving: The trained model, along with the preprocessing steps, is persisted using joblib to a file named model.pkl for future use and deployment.

By following this methodology, accurate predictions regarding reservoir storage levels can be obtained, facilitating better water resource management and decision-making processes.

Scheme	Scheme	Design C	FRL (m)	Rule Lev	Present \	Present (	Percent	Inflow (Cu	Outflow F	Outflow (	Cumm. R	Type	Gate Pos	Opening(m)
36	Ukai	7414.3	105.16	105.16	105.01	7328.8	98.85	18261	0	800	1431	G	0	0
42	Damang	524.86	79.86	79.86	79.86	524.86	100	4265	4385	75	2485.8	G	2	0.3
27	Watrak	158.2	136.25	136.25	133.87	103.76	65.59	0	0	0	842	G	0	0
28	Guhai	68.75	173	173	170.25	38.39	55.85	82	0	0	772	G	0	0
23	Mazam	43.86	157.1	157.1	154.28	20.89	47.64	25	0	0	875	G	0	0
30	Hathmati	152.93	180.75	0	178.64	74.18	48.5	0	0	0	797	UG	0	0
32	Javanpu	2.5	91	91	91	2.5	100	180	180	0	755	G	1	0.1
33	Harnav-I	21.67	332	332	331.5	20.52	94.72	0	0	0	728	G	0	0
34	Meshwo	53.13	214.59	0	210.36	28.82	54.25	80	0	0	740	UG	0	0
12	Wanakb	41.88	67.3	0	68.12	46.61	100	16038	13588	2750	1006.6	UG	0	0
14	Panam	578.19	127.41	127.41	127.41	578.18	100	2901	2852	200	939	G	1	0.6
16	Hadaf	22.09	166.2	166.2	166.2	22.09	99.98	690	681	0	700	G	1	0.15
17	Kadana	1249.3	127.71	127.71	127.66	1244.2	99.59	12119	5100	700	821	G	0	0
6	Karjan	538.75	115.25	115.25	114.24	523.04	97.08	4207	1461	0	878	G	1	0.2
40	Sukhi	173.01	147.82	147.82	147.78	174.3	100	521.95	521.95	0	1172	G	1	0.15
3	Mukteshv	31.46	201.65	201.65	193.8	23.26	73.93	100	0	0	917	G	0	0
4	Dantiwac	397.12	184.1	184.1	184.1	393.62	99.12	935	935	2	825	G	1	0.3
5	Sipu	161.43	186.43	186.24	180.6	53.22	32.97	0	0	0	460	G	0	0
13	Dharoi	813.14	189.59	189.28	189.33	785.92	96.65	2156	1506	650	819.4	G	1	0.3
65	Khodiyar	29.94	202.68	202.68	201.25	22.74	75.97	130	0	0	820	G	0	0
76	Shetrunji	346.48	55.53	55.53	55.53	346.48	100	0	0	0	540	G	0	0
94	Und-I	69.05	98	98	98	69.05	100	321	321	0	844	G	1	0.15
148	Bhadar	188.14	107.9	107.9	107.35	164.35	87.36	95	0	0	785	G	0	0
149	Bhadar -	49	53.1	53.1	53.1	49	100	0	0	0	825	G	0	0
150	Machchl	87.9	57.3	57.3	56.12	67.21	76.46	430	0	0	589	G	0	0
158	Machchl	68.95	135.33	0	135.02	61.17	88.71	0	0	0	625	UG	0	0

Fig. 3 Pre-processed Dataset used

1. Support Vector Machine

The Support Vector Machine (SVM) is a supervised learning model utilized for classification and regression tasks. It employs the kernel trick to transform data and establishes an optimal boundary, known as a hyperplane, to distinguish between different outputs based on these transformations. This boundary is determined using support vectors, which are the points closest to the line [11]. The margin, defined as the distance between these support vectors and the line, plays a crucial role, with a wider margin indicating better performance. However, SVM's inherent linearity can be seen as a limitation, particularly for nonlinear data, as it may affect its effectiveness in such cases.

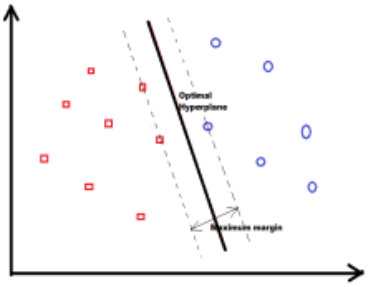

Fig 4. Working of SVM

```
Accuracy: 0.9411764705882353
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.83      0.91         6
           1       0.92      1.00      0.96        11

    accuracy                           0.94        17
   macro avg       0.96      0.92      0.93        17
weighted avg       0.95      0.94      0.94        17
```

Fig. 5 Classification Report of SVM

2. Logistic Regression

Logistic regression is a supervised learning algorithm commonly used for classification purposes, especially when dealing with categorical and labeled target variables. Its simplicity lies in its utilization of a basic regression formula for prediction, followed by the application of a

threshold (typically 0.5) to classify the predicted value into either category 0 or category 1.

$$y = mx + b.$$

Accuracy: 0.7647058823529411
Classification Report:

	precision	recall	f1-score	support
0	0.60	1.00	0.75	6
1	1.00	0.64	0.78	11
accuracy			0.76	17
macro avg	0.80	0.82	0.76	17
weighted avg	0.86	0.76	0.77	17

Fig. 6 Classification Report of Logistic Regression

3. Random Forest Regression

Random Forest is a supervised classification algorithm that is applicable to both classification and regression problems. Its advantages encompass the ability to handle missing values, model classification for categorical values, and mitigate overfitting even when utilizing a larger number of trees created in the forest [3]. The mathematical formula for Random Forest can be expressed as:

$ni_j = W_j C_{j-left(j)} C_{left(j)} - W_{right(j)} C_{right(j)}$

Accuracy: 0.9411764705882353
Classification Report:

	precision	recall	f1-score	support
0	1.00	0.83	0.91	6
1	0.92	1.00	0.96	11
accuracy			0.94	17
macro avg	0.96	0.92	0.93	17
weighted avg	0.95	0.94	0.94	17

Fig. 7 Classification Report of Random Forest Regression

4. Decision Tree

The Decision Tree is a supervised machine learning algorithm commonly used for classification purposes. It operates on a tree-like structure, where each node represents a test on a particular attribute, each branch denotes an outcome from that test, and each leaf node indicates the class label associated with the path it leads to [3]. The importance of a node in decision-making decreases as we move down the tree [3]. Different Decision tree techniques used are :

1) CART (Classification and Regression Tree) which uses the Gini index as a metric.

Formula for Gini Index:

$$\text{gini}_A(D) = \frac{|D1|}{D}\,\text{gini}(D_1) + \frac{|D2|}{D}\,\text{gini}(D_2)$$

2) ID3 (Iterative Dichotomiser which uses Entropy Function and Information Gain as metrics).

Formula for Information Gain:

$$I(P_i, N_i) = \frac{-p}{p+n} log_2 \frac{p}{p+n} - \frac{n}{p+n} log_2 \frac{n}{p+n}$$

Formula for Entropy:

$$\Sigma \left(\frac{Pi+Ni}{p+n}\right) I(P_i, N_i)$$

Accuracy: 0.9411764705882353
Classification Report:

	precision	recall	f1-score	support
0	1.00	0.83	0.91	6
1	0.92	1.00	0.96	11
accuracy			0.94	17
macro avg	0.96	0.92	0.93	17
weighted avg	0.95	0.94	0.94	17

Fig. 8 Classification Report of Decision Tree

5. Naive Bayes

Naive Bayes is a probabilistic machine learning algorithm widely applicable in various classification tasks. It stands out for its ease of implementation and quick prediction capabilities, owing to its probabilistic nature. Based on Bayes' theorem, it excels particularly in scenarios with high-dimensional inputs. The mathematical formula for the theorem is represented as follows:

$$P(A|B)=P(B|A).P(A)/P(B)$$

Here A and B are two events and P(A/B) is the conditional probability that event A occurs, given that event B has occurred. P(B|A) is the conditional probability that event B occurs, given that event A has occurred. P(A) and P(B) : Probability of A and B without regard to each other.

The Naive Bayes model offers numerous advantages, such as straightforward probabilistic predictions, fast computation for both training and prediction, and easy interpretation.

Accuracy: 0.9411764705882353
Classification Report:

	precision	recall	f1-score	support
0	0.86	1.00	0.92	6
1	1.00	0.91	0.95	11
accuracy			0.94	17
macro avg	0.93	0.95	0.94	17
weighted avg	0.95	0.94	0.94	17

Fig. 8 Classification Report of Naive Bayes

6. LSTM Neural Network

A conventional RNN operates with a single hidden state passed through time, posing challenges in learning

long-term dependencies. To overcome this, LSTM (Long Short-Term Memory) networks introduce a memory cell, capable of retaining information over extended periods. LSTM networks efficiently capture prolonged dependencies within sequential data, making them well-suited for tasks such as language translation, speech recognition, and time series forecasting[9] [10].

Activation at time *t*:

$$h_t^j = (1 - z_t^j)h_{t-1}^j + z_t^j \tilde{h}_t^j$$

Update gate:

$$z_t^j = \sigma(W_z x_t + U_z h_{t-1})^j$$

Candidate activation:

$$\tilde{h}_t^j = tanh(Wx_t + U(rt \otimes h_{t-1}))^j$$

Reset gate:

$$r_t^j = \sigma(W_r x_t + U_r h_{t-1})^j$$

Where:

Activation at time *t* (h_t^j)

The hidden state at time *t* for the *j-th* unit or neuron in the LSTM network.

Update gate(z_t^j)

The update gate at time *t* for the *j-th* unit or neuron in the LSTM network. It controls how much of the previous cell state (h_{t-1}^j) is retained and how much of the new candidate value ($\tilde{h}_t^j$) is added to the current cell state.

Candidate activation($\tilde{h}_t^j$):

The candidate activation at time *t* for the *j-th* unit or neuron in the LSTM network. It represents the new information that could be added to the cell state at time t.

Reset gate (r_t^j):

The reset gate at time *t* for the *j-th* unit or neuron in the LSTM network. It controls how much of the previous hidden state (h_t^j) is forgotten or reset when computing the candidate activation ($\tilde{h}_t^j$)

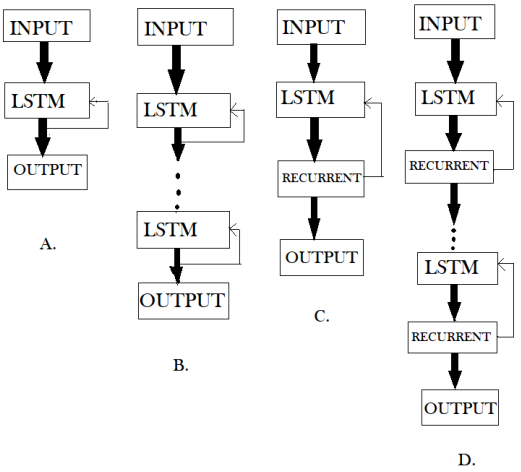

Fig 9. Working of LSTM

Accuracy: 0.8823529411764706
Classification Report:

	precision	recall	f1-score	support
0	0.83	0.83	0.83	6
1	0.91	0.91	0.91	11
accuracy			0.88	17
macro avg	0.87	0.87	0.87	17
weighted avg	0.88	0.88	0.88	17

Fig. 10 Classification Report of LSTM

7. Prediction Using RNN

A Recurrent Neural Network (RNN) is a specialized algorithm endowed with internal memory, enabling it to store information from previous inputs, making it particularly adept at processing sequential data [3]. Unlike Feed Forward Neural Networks, which allow information to flow only in one direction, RNN incorporates feedback loops, allowing it to consider both the current input and the knowledge gleaned from past inputs when making decisions [3]. This unique characteristic distinguishes RNN from other algorithms. RNN assigns weights to both the current input and the previous input, iteratively adjusting these weights through gradient descent and backward propagation [3]. The mathematical formulations for RNN are as follows:

$o^t = f(h^t, \theta)$
$h^t = k(h^{t-1}, x^t, \theta)$
Where, o^t is the output produced at time t,
h^t is the state of hidden layers at time t,

x^t is the input given at time t,
θ indicates the weights and biases of that particular network

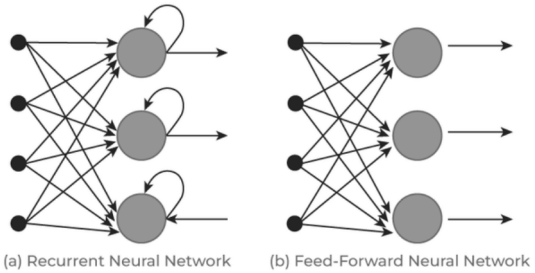

Fig. 11 Working of Recurrent Neural Network

```
Accuracy: 0.7058823529411765
Classification Report:
              precision    recall  f1-score   support

           0       0.67      0.33      0.44         6
           1       0.71      0.91      0.80        11

    accuracy                           0.71        17
   macro avg       0.69      0.62      0.62        17
weighted avg       0.70      0.71      0.67        17
```

Fig. 12 Classification report of Recurrent Neural Network

VII. RESULTS & DISCUSSION

We have developed a website using Python Flask. A nodal officer logs in using their credentials and is shown only the schemes under their supervision. They input the real-time data of the scheme and get the prediction of whether the water level is safe or an overflow will occur. In both the cases, a notification via SMS will be sent to the desired user (i.e. the villagers living on the riverbank of that scheme)

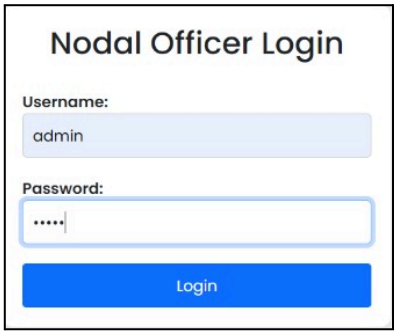

Fig. 13 Nodal Officer Login

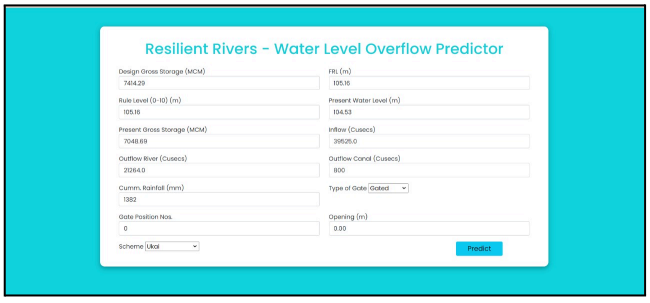

Fig. 14 Predictor page

Fig. 14 SMS sent via API

VIII. CONCLUSION

The proposed reservoir flood warning system uses advanced flood prediction models and real-time data integration to issue timely alerts and warnings.The system aims to enhance disaster preparedness and minimize flood risks.The system seeks to create a resilient approach to flood management.The successful implementation of the system will contribute to improved public safety, disaster response, and sustainable water resource planning.

IX. FUTURE WORK

The system will be deployed on a website and will be available for the nodal officer of the reservoir or dam.The system can be integrated with other state-specific systems to share information and ensure everyone is on the same page.A mobile app may be developed to allow users to receive flood warnings and information on their smartphones.The system will be used to collect data on flood events and their impacts to improve flood prediction models.The system will be used to educate the public about flood risk and preparedness through public awareness campaigns and school programs.

REFERENCES

[1] Zhaoli Wang , Chengguang Lai , Xiaohong Chen , Bing Yang , Shiwei Zhao , Xiaoyan Bai, Flood hazard risk assessment model based on random forest, Year of Publication: August 2015.

[2] Sunmin Lee, Jeong-Cheol Kima, Hyung-Sup Junga, Moung Jin Leecand Saro Lee, Spatial prediction of flood susceptibility using random-forest and boosted-tree models in Seoul metropolitan city, Korea, Year of Publication: 10 Apr 2017.

[3] Chinmayee Kinage, Sejal Mandora, Abhishek Kalgutkar, Sunita Sahu , Amruta Parab, Performance Evaluation of Different Machine Learning Based Algorithms for Flood Prediction and Model for Real Time Flood Prediction, Year of Publication: 2019, 5th International Conference On Computing, Communication, Control And Automation (ICCUBEA).

[4] Nikunj K. Mangukiya; Darshan J. Mehta; Raj Jariwala, Flood frequency analysis and inundation mapping for lower Narmada basin, India, Year of Publication: 1 February 2022.

[5] Minister of State for Jal Shakti, Shri Bishweswar Tudu, Flood Forecasting and Early Warning System, Publication Date : 08 AUG 2022 6:02PM by PIB Delhi

[6] Press Information Bureau, Government of India, Ministry of Water Resources, River Development and Ganga Rejuvenation, Flood Control Schemes by Indian Government, Year of Publication: January 10, 2023

[7] Darshan Mehta , Jay Dhabuwala , Sanjaykumar M. Yadav , Vijendra Kumar , Hazi M. Azamathulla , Improving flood forecasting in Narmada river basin using hierarchical clustering and hydrological modeling, Year of Publication: December 2023

[8] A Yovan Felix , T Sasipraba , Flood Detection Using Gradient Boost Machine Learning Approach , Year of Publication: December 2019 , Conference: 2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)

[9] Jiaojiao Hu, Xiaofeng Wang, Ying Zhang, Depeng Zhang, Meng Zhang, Jianru Xue , Time Series Prediction Method Based on Variant LSTM Recurrent Neural Network ,Neural Processing Letters, 2020

[10] Pin Zhang , Zhen-Yu Yin , State of the-Art Review of Machine Learning Applications in Constitutive Modeling of Soils, Archives of Computational Methods in Engineering, 2021

[11] Yutaro Fuse, Yoshitaka Nagashima, Hiroshi Nishiwaki, Fumiharu Ohka et al., Development of machine learning models for predicting unfavorable functional outcomes in patients with chronic subdural hematomas , Research Square Platform LLC, 2022