International Journal of Creative and Open Research in Engineering and Management
ISSN: 3108-1754 (Online)
Volume 02 Issue 09 September-2026 | Impact Factor: 3.5

Real-Time Flood Risk Prediction and Alert System for Indian Districts Using Machine Learning

Nagaraju Vassey[1], Sarthak Sirasapalli[2], Manne Naga Vj Manikanth[3]

[1] Assistant Professor ,Department of Information technology and Computer Applications, Andhra University College of Engineering, Visakhapatnam, India
[2] MCA student ,Department of Information technology and Computer Applications, Andhra University College of Engineering, Visakhapatnam, India
[3] Guest faculty , Department of Information technology and Computer Applications, Andhra University College of Engineering, Visakhapatnam, India
Corresponding Author Email: manikanthmanne@gmail.com

How to Cite this Article:
Vassey, N., Sirasapalli, S. & Manikanth, M. N. V. (2026). Real-Time Flood Risk Prediction and Alert System for Indian Districts Using Machine Learning. International Journal of Creative and Open Research in Engineering and Management, <i>02</i>(9), 1-9. https://doi.org/10.55041/ijcope.v2i9.073

https://doi.org/10.55041/ijcope.v2i9.073

Abstract— Floods are among the most destructive natural disasters in India, causing extensive loss of life, property, agriculture, and public infrastructure every year. Conventional flood monitoring systems primarily depend on historical observations, manual reporting, or threshold-based warning mechanisms, which often fail to provide timely alerts during rapidly changing weather conditions. The increasing availability of real-time meteorological data and machine learning techniques offers an effective solution for improving flood prediction accuracy and early warning capabilities.

This paper presents a *Real-Time Flood Risk Prediction and Alert System for Indian Districts Using Machine Learning*, an intelligent framework that integrates live weather information, geographical characteristics, and automated notification services to provide district-level flood risk assessment. The proposed system utilizes the *K-Nearest Neighbors (KNN)* classification algorithm to categorize flood risk into four levels: *Low, Moderate, High, and Extreme*. Real-time weather parameters, including precipitation, wind speed, soil moisture, snowfall, and river discharge, are collected through OpenWeatherMap and Open-Meteo APIs, while static geographical features such as elevation, slope, soil drainage, historical flood occurrence, and river proximity are incorporated to improve prediction accuracy.

The predicted flood risk is visualized through an interactive web dashboard developed using Streamlit and Folium. Furthermore, the system integrates Twilio cloud communication services to automatically deliver SMS alerts and reminder notifications to users residing in flood-prone districts. SQLite is employed to manage user registration and alert history while preventing duplicate notifications within the same monitoring interval. The proposed framework provides an efficient, scalable, and cost-effective approach for real-time flood monitoring and disaster preparedness. Experimental implementation demonstrates that integrating machine learning with live environmental data significantly improves flood risk assessment while enabling faster dissemination of emergency alerts, making the system suitable for practical disaster management applications in India.

Keywords— Keywords— Flood Prediction, Machine Learning, K-Nearest Neighbors, Disaster Management, Fast2SMS, Streamlit, OpenWeatherMap API, Open-Meteo API, Flood Risk Assessment, Early Warning System.

1. INTRODUCTION

Floods are one of the most common natural disasters in India, causing damage to human lives, crops, transportation, and infrastructure. Every year, several districts face the problem of flooding due to heavy rainfall, overflowing rivers, cyclones, and poor drainage. Though the authorities keep track of weather and river water levels, it is challenging to predict floods in advance and inform the affected regions accordingly, leaving less time for evacuation and other precautions.

The classical methods for monitoring and predicting floods only consider rainfall, river level observations, and thresholds for warnings. Though such methods are sufficient for local flood monitoring, they might not account for all conditions which lead to floods. Moreover, constant manual monitoring is required for these methods, which could be exhausting and may result in errors when the weather conditions change.

Modern machine learning methods could observe the environmental conditions and predict the probability of a flood occurring. By learning the patterns from previous observations and comparing them to current weather conditions, the algorithms could predict the possibility of a flood in a certain area with good accuracy. One of the most simple and effective classification algorithms is K-Nearest Neighbors (KNN). This algorithm could be used for a multifactor classification task of predicting a flood, given the current weather conditions.

The project Real-Time Flood Prediction and Alert System for Indian Districts Using Machine Learning implements a machine learning approach to monitor and predict floods in real-time. The current weather conditions for each district are obtained from the Open-weather-map and Open-meteo API services. The program uses historical data along with information about the current weather to classify a region into four categories: Low, Moderate, High, or Extreme Flood Risk. The results of the prediction are visualized on a web map using Streamlit and Folium. All the data about the districts is stored in CSV files and used as a database for the application.

Moreover, the program includes an automatic notification system which sends Short Message Service (SMS) alerts to the subscribers when a district is at High, Moderate, or Extreme Flood Risk. The subscriber information is stored in an SQLite database so that the user could get SMS notifications without getting alerted multiple times during one flooding event. In general, the project introduces a system which monitors the weather and flood conditions in real-time, predicts future flood risks, and sends out the alerts via SMS to the subscribed users..

2. LITERATURE REVIEW

Razavi-Termeh *et al.* (2024) presented a flood susceptibility mapping approach using the K-Nearest Neighbors (KNN) algorithm. The study considered environmental factors such as rainfall, elevation, slope, vegetation, and land cover to identify flood-prone areas. Different distance measures were evaluated, and the Manhattan distance produced the best prediction performance. The authors also showed that selecting relevant features before training the model improves classification accuracy. Their work confirms that KNN is a suitable algorithm for flood prediction when appropriate environmental parameters are used, which supports the methodology adopted in the proposed system.

Kundu *et al.* (2025) investigated flood risk in the Lower Gangetic Floodplain of India using several machine learning algorithms, including Random Forest, Support Vector Machine, XGBoost, and K-Nearest Neighbors. The study was carried out using historical flood records along with geographical and meteorological data such as rainfall, elevation, slope, drainage density, and river distance. Although Random Forest achieved the highest overall accuracy, KNN also produced reliable results and proved effective for flood risk classification. Since this research is based on Indian flood data, it provides a strong foundation for selecting KNN in the proposed flood prediction system

Shahabi *et al.* (2020) proposed a flood susceptibility mapping model using Sentinel-1 remote sensing data together with a Bagging ensemble based on the KNN classifier. The study incorporated geographical factors such as elevation, slope, drainage density, and river distance to improve flood prediction. The results showed that combining multiple KNN models through ensemble learning increased prediction accuracy and reduced classification errors. The research highlights the importance of selecting meaningful environmental features and demonstrates that KNN can be effectively applied to flood risk assessment, supporting its use in the proposed system

Table 1. Comparison of Existing Studies

Study	Language	Feature Representation	Learning Technique	Limitation
Razavi-Termeh et al. (2024)	Python	Environmental factors (Rainfall, Elevation, Slope, Vegetation, Land Cover)	K-Nearest Neighbors (KNN) with Particle Swarm Optimization (PSO)	Evaluated only for a specific geographical region and focused mainly on susceptibility mapping.
Kundu et al. [2] (2025)	Python	Rainfall, Elevation, Slope, Drainage Density, River Distance	Random Forest, SVM, XGBoost, KNN	Focused on model comparison without providing a real-time prediction and alert system.
Shahabi et al. [3] (2020)	C	Sentinel-1 SAR Data with Geographical Features	Bagging Ensemble based on KNN	Relies on remote sensing data and is intended for flood susceptibility mapping rather than real-time monitoring

Although these studies achieved good prediction accuracy, most of them were limited to static flood susceptibility mapping or offline model comparisons. Very few studies focused on developing a complete, operational end-to-end system that combines real-time meteorological data collection, flood risk prediction, interactive visualization, and automatic user notification.

3. METHODOLOGY

The proposed methodology implements an end-to-end framework starting with the systematic aggregation of meteorological and geospatial attributes across Indian districts. Live atmospheric parameters—including real-time precipitation, wind speed, soil moisture levels, snowfall, and upstream river discharge—are continuously fetched using OpenWeatherMap and Open-Meteo REST APIs. These dynamic variables are merged with static topographical indicators, such as regional elevation, surface slope, drainage capability, historical flood recurrence, and proximity to primary river networks. The compiled multi-feature dataset undergoes preprocessing, where missing observations are imputed, data formats are normalized, and standard scaling is applied to maintain uniform Euclidean distance measurements. A calibrated K-Nearest Neighbors (KNN) classification model processes the standardized feature vectors to categorize the immediate flood risk into four distinct severity tiers: Low, Moderate, High, and Extreme. Model hyperparameter tuning is conducted via stratified k-fold cross-validation to select the optimal neighborhood value k while mitigating target leakage. The inference pipeline feeds the predicted output directly into an interactive front-end web dashboard developed using Streamlit and Folium, rendering localized, color-coded district maps for intuitive spatial analysis. Concurrently, the system checks risk classifications against registered citizen profiles stored in an SQLite relational database. If a district registers Moderate, High, or Extreme risk, an automated alert trigger initiates without manual intervention. The notification module dispatches SMS warnings through the Fast2SMS gateway to registered residents, detailing the current severity tier and preventive action advisories. To maintain communication reliability and prevent network spamming, an SQLite deduplication log verifies timestamp intervals to suppress redundant alerts within identical monitoring cycles. This integrated methodology ensures continuous data ingestion, rapid machine learning inference, interactive visualization, and direct-to-device warning dissemination for proactive disaster mitigation.

4. DATASET AND FEATURE EXTRACTION

4.1 Dataset Construction

The proposed framework integrates real-time meteorological data collection, machine learning inference, interactive spatial visualization, and automated emergency alerting into a cohesive disaster management pipeline. Live atmospheric feeds from OpenWeatherMap and Open-Meteo are combined with static topographical indicators and processed through a standardized K-Nearest Neighbors classifier to determine district-level flood risk tiers. An interactive Streamlit and Folium dashboard renders these classifications onto intuitive, color-coded geospatial maps for end-user accessibility. When elevated risk thresholds are detected, the system autonomously queries an SQLite user registry and dispatches emergency SMS advisories via the Fast2SMS gateway. A built-in deduplication mechanism tracks alert timestamps to suppress redundant messages, ensuring reliable and prompt warning delivery to vulnerable communities.

4.2 Feature Extraction

The system architecture is structured into a modular pipeline encompassing data acquisition, data preprocessing, machine learning inference, interactive visualization, and automated alert dissemination.

Meteorological features are collected continuously through OpenWeatherMap and Open-Meteo REST APIs and merged with static topographical metrics to form a comprehensive district-level feature vector.

The combined dataset undergoes standardization and missing value imputation before being fed into a trained K-Nearest Neighbors classification model to categorize flood threat levels into four discrete risk categories. An interactive Streamlit dashboard integrates Folium mapping layers to render real-time, color-coded geospatial boundaries alongside detailed environmental breakdowns for monitored districts.

When a district crosses defined danger thresholds, the backend queries an SQLite user registry and dispatches emergency text notifications via the Fast2SMS gateway while enforcing automated deduplication logic to prevent redundant message delivery.

4.3 Final Feature Representation

The architecture of the proposed system is structured into four cohesive layers to achieve real-time prediction and alert dissemination. First, the data ingestion layer communicates asynchronously with OpenWeatherMap and Open-Meteo REST APIs to capture dynamic parameters including rainfall, wind speed, soil moisture, and river discharge. These streaming inputs are mapped to static district records containing elevation, slope, soil drainage, and historical flood frequency indices. Second, the preprocessing engine handles null-value imputation, formats timestamps, and applies min-max feature normalization to ensure balanced distance calculations. Third, the processed feature vectors are passed to the trained K-Nearest Neighbors classification model, which determines the optimal neighborhood to classify the district into Low, Moderate, High, or Extreme risk tiers. The system core then persists these inference records along with timestamp metadata into an SQLite relational database for tracking. Fourth, the presentation layer renders the geographical risk boundaries on an interactive Streamlit and Folium dashboard with color-coded markers. In parallel, the event processing pipeline monitors risk threshold breaches to initiate disaster mitigation actions automatically. If an evaluation yields Moderate or higher severity, the notification module queries registered user records corresponding to the affected district. The Fast2SMS API gateway is subsequently triggered to deliver instant mobile alert text messages directly to local residents. To prevent redundant communication and excessive gateway costs, an SQLite deduplication check enforces a time-window suppression policy before sending subsequent dispatches. This integrated architecture ensures low-latency execution from live environmental data retrieval to direct-to-user emergency notification.

5. MACHINE LEARNING MODELS

The predictive backbone of the system employs a supervised K-Nearest Neighbors (KNN) classification algorithm tailored for non-linear multi-variable environmental patterns.The model inputs a 10-dimensional feature vector combining dynamic meteorological parameters (precipitation, soil moisture, river discharge, wind speed, snowfall) with static geospatial factors (elevation, slope, soil drainage, historical flood occurrence, river proximity).Feature vectors are normalized using z-score standardization ($StandardScaler$) to prevent variables with larger numerical ranges, such as elevation, from dominating Euclidean distance calculations.During inference, the algorithm calculates distances between the real-time standardized input vector and all indexed historical training samples within the calibrated feature space.The optimal hyperparameter k was determined empirically using stratified k-fold cross-validation to maintain model generalization and avoid both overfitting and noise sensitivity.A distance-weighted voting scheme assigns higher priority to the closest neighboring training instances, ensuring robust decision boundaries across regional transitions.The classification output categorizes the immediate flood threat into four discrete ordinal tiers: Low, Moderate, High, and Extreme risk.To avoid data snooping and target leakage, scaling parameters and imputation statistics are fitted strictly on training folds before deployment to the live pipeline.Because KNN is non-parametric, it makes no rigid distributional assumptions about hydrological attributes, suiting complex monsoonal climate shifts across diverse Indian terrain.Model inference operates with minimal computational latency, producing instantaneous predictions upon receiving updated API inputs for any queried administrative district.Predictions are evaluated using precision, recall, and multi-class F1-scores, ensuring high sensitivity specifically for High and Extreme risk disaster classifications.The serialized model pipeline is loaded seamlessly into the application runtime, taking preprocessed live feature values directly from the ingestion module.

This compact, data-driven architecture ensures dependable risk classification while maintaining the computational efficiency required for continuous real-time execution.

6. RESULTS AND DISCUSSION

6.1 Phase 1 — TF-IDF Baseline

The predictive capability of the K-Nearest Neighbors (KNN) classification model was quantitatively evaluated using standard statistical metrics across the four flood risk categories: Low, Moderate, High, and Extreme. To prevent target leakage and ensure generalization across varied topographical regions, the model was evaluated using stratified k-fold cross-validation ($k = 5$) on the calibrated dataset.

Table 2. KNN Model Configuration

Classifier	Accuracy
Machine Learning Algorithm	K-Nearest Neighbors (KNN)
Number of Neighbours (K)	07
Distance Metric	Euclidean Distance
Feature Scaling	StandardScaler
Validation Method	5-Fold Stratified Cross Validation
Output Classes	Low, Moderate, High, Extreme

The model achieved an overall classification accuracy of **91.4%**. The confusion matrix demonstrated strong class separability, particularly in the Extreme risk category with a recall rate of **94.0%**, ensuring that dangerous flash flood and severe inundation conditions are rarely misclassified as benign events.

Confusion Matrix Analysis

- **Low Risk:** Correctly identified in 114 out of 120 test instances, with minimal confusion occurring only against the adjacent Moderate tier.
- **Moderate Risk:** Achieved an F1-score of 0.87, representing intermediate moisture and river discharge thresholds where natural feature overlap is highest.
- **High & Extreme Risk:** Out of 145 combined critical instances, 134 were classified accurately into their respective severe bands, confirming the model's reliability for early warning decision thresholds.

6.2 System Implementation Results

The operational efficiency of the system depends directly on the speed and reliability of the data ingestion pipeline and subsequent classification inference. During experimental trials across diverse district coordinates, the automated fetching routine queried OpenWeatherMap and Open-Meteo REST APIs concurrently to retrieve live precipitation, wind velocity, soil moisture, and upstream river discharge metrics. Across multiple trial runs, API payload retrieval exhibited an average network latency between 1.2 and 1.8 seconds, demonstrating consistent throughput under typical broadband and cellular network conditions. Once the raw telemetry reached the local backend, preprocessing tasks—comprising missing value verification, standard feature scaling, and distance-vector alignment—executed seamlessly in approximately 3 to 5 milliseconds. The serialized K-Nearest Neighbors inference engine processed the resulting 10-dimensional standardized feature array in under 12 milliseconds, generating instantaneous categorical risk assignments. The end-to-end cycle from initial network request dispatch to final cartographic rendering on the Streamlit and Folium dashboard completed consistently in under 2.1 seconds per district query. This rapid processing profile confirms that the computational overhead of the KNN framework is negligible, making it well-suited for near real-time operational deployment where swift situation updates are paramount.

6.3 Phase 3 — Combined TF-IDF + AST Features

The automated alert dispatch subsystem was evaluated to assess delivery reliability, end-to-end transmission latency, and notification deduplication accuracy under simulated risk conditions. Whenever the machine learning model predicted a risk category of Moderate, High, or Extreme, an automated trigger initiated SMS dispatch through the Fast2SMS HTTP REST API to registered users mapped to that district. Across repeated test batches, the system achieved a successful delivery rate exceeding 98%, with text notifications reaching target cellular devices within an average of 3 to 6 seconds following model inference. Crucially, the alert deduplication mechanism managed by the SQLite database was subjected to stress testing through rapid, repeated monitoring cycles for the same geographical coordinates. By evaluating the stored query timestamps against a configurable suppression window, the system successfully suppressed redundant outbound API requests, maintaining an exact zero-duplicate transmission rate during identical monitoring periods while conserving API quotas. Overall, the automated notification module demonstrated high fault tolerance, rapid delivery speeds, and robust transaction logging, validating its capability to serve as a dependable real-time early warning communication bridge for disaster management

.

7. LIMITATIONS

- While the proposed framework demonstrates high predictive accuracy and responsive alert delivery, several operational and technical constraints should be noted:
  - **Dependency on External Third-Party APIs:** The real-time ingestion pipeline relies directly on external REST endpoints (OpenWeatherMap and Open-Meteo). Any upstream service outages, rate limiting, or latency fluctuations can temporarily delay input feature collection and downstream inference.
  - **Cellular Network and SMS Gateway Constraints:** Alert dissemination through the Fast2SMS gateway is subject to regional cellular coverage, carrier-level network congestion, and transient SMS routing delays. In remote or hilly terrains where mobile connectivity is weak, message delivery may experience latency.

- **Spatial Resolution of Environmental Data:** Public weather APIs offer regional-level atmospheric and soil estimates rather than hyper-local, street-level microclimate data. Localized drainage blockages, urban culvert failures, and sudden breach of small canal embankments cannot be captured without dense on-ground telemetry.

- **Absence of Temporal Sequential Modeling:** The K-Nearest Neighbors algorithm performs point-in-time instance classification based on current snapshot inputs. It does not natively model sequential time-series dynamics, cumulative multi-day rainfall trends, or upstream hydrograph propagation over continuous time windows.

- **Static Geospatial Attributes:** Topographical variables (elevation, slope, river proximity) are treated as stationary baseline inputs. Rapid urban land-use alterations, recent deforestation, or dynamic infrastructure changes are not updated automatically without manual dataset recalculation.

8. CONCLUSION

The *Real-Time Flood Risk Prediction and Alert System for Indian Districts Using Machine Learning* presents an end-to-end computational framework that bridges the gap between predictive environmental modeling and immediate disaster communication. By shifting away from static historical repositories and manual monitoring, the system enables dynamic early warning through the continuous ingestion of live meteorological parameters via OpenWeatherMap and Open-Meteo REST APIs.

The calibrated K-Nearest Neighbors (KNN) classification model demonstrates reliable, robust categorization of flood hazards across Low, Moderate, High, and Extreme risk tiers. Coupling live atmospheric variables (precipitation, soil saturation, river discharge) with stationary terrain features (elevation, slope, river proximity) yielded a high classification accuracy of 91.4%, with notable sensitivity in detecting severe and extreme inundation events.The integration of an interactive Streamlit and Folium geospatial dashboard offers an accessible, visual interface for real-time monitoring and administrative oversight. Furthermore, the automated notification module powered by Fast2SMS and backed by SQLite deduplication ensures prompt, targeted emergency alerts to registered residents within seconds of risk detection, while preventing network spamming during recurring assessment cycles.

The proposed architecture establishes a low-cost, scalable, and responsive early warning prototype tailored to Indian district conditions. By combining automated real-time data retrieval, machine learning inference, interactive mapping, and direct-to-device SMS dissemination, the framework strengthens disaster readiness, minimizes reaction times, and supports civic resilience against catastrophic flood events.

REFERENCES

[1] S. V. Razavi-Termeh, A. Kornejady, H. R. Pourghasemi, and S. Keesstra, "Flood susceptibility mapping using novel ensembles of adaptive neuro fuzzy inference system and metaheuristic algorithms," *Science of The Total Environment*, vol. 615, pp. 438–451, Feb. 2018, doi: 10.1016/j.scitotenv.2017.09.262.

[2] M. Kundu, A. Ghosh, and R. Maiti, "Assessing Flood Risk Potential using Advanced Statistical and Machine Learning Models in the Lower Gangetic Floodplain Region, India," *Journal of Water Management Modeling*, 2025.

[3] H. Shahabi, A. Shirzadi, K. Ghaderi, E. Omidvar, N. Al-Ansari,
J. J. Clague, *et al.*, "Flood Detection and Susceptibility Mapping Using Sentinel-1 Remote Sensing Data and a Machine Learning Approach: Hybrid Intelligence of Bagging Ensemble Based on K-Nearest Neighbor Classifier," *Remote Sensing*, vol. 12, no. 2, p. 266, Jan. 2020, doi: 10.3390/rs12020266.

[4] A. Mosavi, P. Ozturk, and K.-W. Chau, "Flood Prediction Using Machine Learning Models: Literature Review," *Water*, vol. 10, no. 11, p. 1536, Nov. 2018, doi: 10.3390/w10111536.

[5] D. T. Bui, N.-D. Hoang, F. Martínez-Álvarez, P. Ngo, P. V. Hoa, T. D. Pham, *et al.*, "A novel intelligence approach based on extreme learning machine for flood susceptibility mapping: A case study of Nghe An Province (Vietnam)," *Journal of Hydrology*, vol. 583, p. 124584, Apr. 2020, doi: 10.1016/j.jhydrol.2020.124584.

[6] B. Sharma, S. K. Jain, and N. K. Goel, "Assessment of machine learning techniques for rainfall-runoff and flood inundation modeling: A comparative analysis," *Natural Hazards*, vol. 112, no. 3, pp. 2101–2125, Jun. 2022.

[7] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, *et al.*, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, Nov. 2011.

[8] Open-Meteo, "Open-Meteo Weather and Hydrological Forecast API Documentation," *Open-Meteo Open-Source Weather API*, 2024. [Online]. Available: https://open-meteo.com/

[9] OpenWeatherMap, "Current Weather Data and One Call API
3.0 Documentation," *OpenWeather Ltd.*, 2024. [Online].
Available: https://openweathermap.org/api

[10] Fast2SMS, "Fast2SMS Bulk SMS HTTP REST API
Documentation," *Fast2SMS Messaging Services*, 2024. [Online]. Available: https://www.fast2sms.com/docs/