INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 1661-1665
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

DISASTER PREDICTION SYSTEM USING MACHINE LEARNING

Nandani Tiwari[1], Nikhil Mishra[2], Anant Panday[3], Dr. Uday Pratap Singh[4]

[1,2,3]Student Dept. Artificial Intelligence And Data Sciencepoornima Institute Of Engineering And Technology Jaipur, India.

[4]Professor Dept. Artificial Intelligence And Data Science Poornima Institute Of Engineering And Technology Jaipur, India.

tiwarinandani2214@gmail.com, nikhilnm150@gmail.com, anantpandey492@gmail.com, uday.pratap@poornima.org

DOI: https://www.doi.org/10.58257/IJPREMS40108

ABSTRACT

Natural disasters like floods and earthquakes have the potential to bring about huge destruction and loss of life. Precise and timely prediction of these events is essential for successful disaster management and mitigation. This project proposes a Disaster Prediction System based on machine learning algorithms and implemented with the Django web framework. The system relies on historical and real-time data, such as seismic activity, rainfall, and water level records, to predict the probability of disasters.

Backend is developed using Django 5.1.6, a strong and scalable web-based interaction platform. Machine learning models are created with scikit-learn (v1.6.1) and trained on preprocessed and cleaned data using pandas (v2.2.3) and NumPy (v2.2.3). The models are serialized with joblib (v1.4.2) to load and predict efficiently in the live system. The system is designed to support modular integration of ML algorithms and has APIs for data input and prediction in real-time.

This solution is intended to support authorities, organizations, and the public by offering a smart and accessible platform for early warning of disasters and decision support.

Keyword: - Disaster Prediction, Machine Learning, Django, Flood Forecasting, Earthquake Prediction, Scikit-learn, Joblib, Pandas, NumPy, Real-time Prediction, Data Analytics, Early Warning System, Web-based ML Application, Model Deployment, Natural Disaster Management

1. INTRODUCTION

Natural catastrophes such as floods hugely affect human life, infrastructure, agriculture, and the economies of the entire globe. Out of all the natural disasters, floods are one of the most recurring and destructive calamities, especially in coastal and low-lying areas. Being a country affected by seasonal monsoons and unpredictable climatic patterns, the threat of flood is a consistent and recurrent danger in India. With increasing unpredictability due to climate change, it has become a dire necessity to predict and detect flood risks early. This project offers a Flood Prediction System that uses machine learning methods to forecast the possibility and severity of floods in terms of location, month, and past rainfall patterns. The overall aim is to develop an intelligent decision-support tool for citizens and local authorities, facilitating disaster preparedness and mitigation. The system has been implemented using the Flask web framework, selected for its light weight and smooth integration with Python-based ML models. Our backend model employs libraries such as scikit-learn, joblib, pandas, and NumPy, supporting real-time rainfall prediction for chosen subdivisions in India. The output of the prediction is classified into alerts—Normal, Orange, and Red—based on the predicted intensity of rainfall.

The model is learned from past rainfall data from credible sources, and given the inputs (location, year, and month), it makes predictions of the anticipated rainfall in millimeters. A user-friendly interface allows users to easily interact with the system, input their location information, and get actionable flood warnings. The system is designed to improve public safety and disaster response through timely data-driven forecasts

2. LITERATURE REVIEW

Flood forecasting has been a very active research field in last several decades, especially with the inclusion of data-driven methods such as machine learning. Conventional flood forecasting was largely based on hydrological models and meteorological observations, which typically involved sophisticated simulations and large amounts of data. Recent developments in machine learning, however, have allowed researchers to develop more flexible and accurate prediction systems.

- Singh et al. (2020) suggested a rainfall forecasting model based on multiple linear regression methodologies to examine Indian long-term monsoon data. Their model reflected the significance of past weather conditions in forecasting subsequent rainfall patterns.

- Raju and Kumar (2019) used Support Vector Machines (SVM) to classify the flood-prone areas on the basis of topography and rainfall. The model demonstrated high accuracy in the prediction of the zones that are prone to flooding, reflecting the effectiveness of ML classification techniques in flood detection.
- Khan et al. (2021) utilized a deep learning method based on Long Short-Term Memory (LSTM) networks for Bangladesh flood forecasting. Their model utilized time-series rainfall data to generate real-time notifications, showing the significance of temporal data in predicting.
- Roy and Dey (2020) applied Random Forest regression to predict rainfall in Indian subdivisions with a high level of precision when forecasting the amount of rainfall. Their contribution favors the employment of ensemble models in regional flood forecasting.

Some web-based flood warning systems have also been created based on Flask and Django frameworks. These systems combine machine learning with real-time visualization, which makes flood prediction tools more accessible and easier to use. Even with these advancements, most current systems are not regionally adaptable or have friendly interfaces. Furthermore, most models are not embedded in a web application that supports real-time interaction and prediction. Our system overcomes these limitations by creating a Flask-based interactive web application that gives rainfall and flood warnings for chosen Indian locations using historical rainfall data. By integrating past datasets with current ML algorithms and ensuring the system is made available via a minimalistic web interface, this project contributes to the expanded body of research in smart disaster prediction systems.

3. RELATED WORKS

Recently, attention had been focused on the application of DRL in algorithmic and quantitative trading as it can tackle tough problems in the financial markets. Currently, many research initiatives are being carried out, but they are unique in methodology or advancement in the implementation of DRL with better trading strategies, risk management, and portfolio optimization. It covers several domains of trading, namely, portfolio optimization, high-frequency trading, and statistical arbitrage within the constraints of standard approaches to machine learning in dynamic, turbulent market conditions. The next section briefly reports on key studies pushing the adoption of DRL in finance from the perspectives of research themes, methodologies, and key findings.

Study	Year	Author(s)	Research Theme	Findings
Flood Prediction Using LSTM	2021	Khan et al.	Time-series deep learning model for flood prediction	LSTM model achieved higher accuracy over traditional methods in forecasting flood events based on historical rainfall data.
Flood Risk Mapping with Random Forest	2020	Patel & Sharma	Ensemble ML-based classificatio n of flood-prone areas	Random Forest and Gradient Boosting correctly classified flood areas based on meteorological and topographic factors.
ANN-based River Level Prediction	2019	Ahmed et al.	Neural network-based water level prediction	ANN predicted river water levels with high precision using historical data, demonstrating strong potential in flood forecasting.
Flood Alert System in Bangladesh	2018	UNDP	Real-time flood monitoring and alert system	Offered early warning based on sensor data, but was not predictive and region-specific.
Rainy Scope Web Application	2020	Independent Developers	Python & Flask-based rainfall visualization tool	Provided simple rainfall statistics by location but not predictions or flood warnings.
FLOODCAST	2022	Open-Source Community	Satellite image-based global flood risk mapping	Hosted satellite and ML models together, but was cloud-dependent and did not offer offline or local model support.

Recent Developments In Flood Forecasting & Disaster Warning Systems

1. Application of Deep Learning Models

- Trend: From Conventional Ml (E.G., Decision Trees) To Deep Learning Models Such As:
- Lstm/Gru For Time-Series Flood Forecasting
- Cnns On Satellite Images For Waterbody Detection
- Why: These Models Learn Intricate Patterns In Rainfall, River Flow, And Soil Data More Effectively Than Traditional Models.

2. Integration of Remote Sensing & Satellite Data

- Trend: Utilizing Sentinel, MODIS, NOAA satellite data to track:
- Rainfall intensity
- Water levels
- Soil moisture
- WHY: ENABLES LARGE-SCALE, REAL-TIME FLOOD MONITORING WITH GREATER ACCURACY.

3. IoT + Sensor-Based Early Warning Systems

- Trend: Low-cost sensors (e.g., Arduino, LoRaWAN) are being used by projects to monitor:
- River water level
- Rainfall
- Soil saturation
- Why: Real-time data enhances the accuracy of early warnings and shortens response time.

4. Geospatial AI (GeoAI)

- Trend: GIS + AI are being used to:
- Predict flood-risk areas
- Create hazard maps
- Optimize evacuation routes
- Why: GeoAI provides spatial smarts and assists authorities to prepare better.

5. Mobile & Cloud-Based Alerting Platforms

- Trend: Utilizing mobile applications and cloud dashboards for:
- Real-time public notifications
- Admin control panels for authorities
- Why: Facilitates mass and rapid communication to users during emergencies.

6. Simulation + ML Hybrid Models

- Trend: Merging physical models (such as HEC-RAS) with ML for:
- Improved flood forecasts
- Real-time weather forecast updates
- Why: Physical + AI modeling enhances reliability across varied landscapes.

7. Explainable AI (XAI)

- Trend: Applying tools such as SHAP or LIME to provide explanations for model predictions
- Why: Facilitates trust-building with disaster management teams by explaining why a prediction was made.

8. Open-Source Collaboration & Data Sharing

- Trend: Governments and NGOs releasing open datasets (e.g., USGS, IMD, NASA)
- Why: Facilitates easier collaborative and accurate model building.

4. BACKGROUND PROBLEM

Floods are one of the most destructive natural disasters that bring widespread damage to life, property, and infrastructure. The effects of climate change have seen the frequency and severity of floods increase drastically in recent years. Developing nations bear the brunt of the loss due to a lack of effective early warning systems, inadequate infrastructure, and response delays.

Conventional flood monitoring systems are mostly based on static thresholds, manual observation, or physical hydrological models that are usually not responsive in real-time, scalable, and predictive in nature. These systems are less appropriate for dynamically changing weather patterns or for early warning based on large-scale environmental data.

Additionally, real-time rainfall, river level, and weather forecasts in most flood-prone locations are either underutilized or not efficiently used. This causes undue delays in sending out warnings, inadequate preparation, and subsequently, more loss of life, damage to livelihoods and ecosystems.

There is a pressing requirement for a data-driven, scalable, and intelligent flood forecasting and alerting system that can

Forecast flooding chances ahead of time with real-time and historical information,

Issue warnings to the community and response teams ahead of time,

Apply current machine learning algorithms to become increasingly accurate over time.

5. METHODOLOGIES

The project adopts a blend of data-driven machine learning, preprocessing methods, and predictive modelling techniques for precise flood forecasting and alert generation. The key methodologies employed are:

1. Data Collection

Sources: Past flood data, rainfall data, river water level data, weather forecasts (e.g., USGS, IMD, local CSV datasets).

Format: Structured CSV files containing relevant flood-related features.

Purpose: Serves as the basis for training and testing predictive models.

2. Data Preprocessing

Cleaning: Dealing with missing or noisy data.

Feature Selection: Determining significant factors (rainfall, temperature, humidity, river level, etc.) that contribute to floods.

Normalization: Normalizing data to enhance model performance.

Time Series Preparation: Organizing data in temporal form for sequence models (if utilized).

3. Machine Learning Modeling

Model Selection: Regression-based models such as Logistic Regression, Random Forest, or even LSTM (in case of time-series).

Training: Past data is divided into training and test sets.

Evaluation Metrics:

- Accuracy
- recision / Recall
- F1-Score
- Confusion Matrix

4. Flood Prediction System

Trained model makes a prediction of the likelihood of flood conditions given current and future inputs.

Output: Binary (Flood / No Flood) or Probability Score.

5. Alert Generation Mechanism

If the model prediction crosses a specific flood risk threshold:

System generates an alert.

Notification may be optionally sent to users or shown on a dashboard.

6. Visualization

Data and predictions may be visualized with:

Matplotlib / Seaborn for plots

GIS maps (optional future work)

7. Deployment (Planned/Future Scope)

Integration with a web or mobile dashboard.

Live updates through cloud-hosted models (e.g., Flask API + Heroku or AWS).

IoT integration for real-time sensor data in later stage.

6. CONCLUSION

The project illustrates the value of machine learning in strengthening natural disaster early warning systems, with a focus on floods. With the use of historical and current environmental data, the system is able to forecast the probability of flooding and create timely warnings to reduce damage and loss of lives. The combination of data preprocessing, intelligent predictive models, and alerting mechanisms provides the basis for a scalable and effective disaster management system. Although the existing model performs encouraging outcomes, more enhancements can be achieved using real-time sensor integration, sophisticated deep learning models, and mobile or cloud platform deployment.

Overall, this work represents a step toward applying data science and AI in constructing more intelligent, responsive, and accessible disaster prediction systems.

7. REFERENCE

[1] USGS Earthquake Hazards Program. United States Geological Survey. https://earthquake.usgs.gov

[2] Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.

[3] Abadi, M., et al. (2016). TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. https://www.tensorflow.org Chollet, F. (2015). Keras. https://keras.io

[4] Van Rossum, G., & Drake, F. L. (2009). Python 3 Reference Manual. Python Software Foundation.

[5] McKinney, W. (2010). Data Structures for Statistical Computing in Python. Proceedings of the 9th Python in Science Conference, 51–56.

[6] Hunter, J. D. (2007). Matplotlib: A 2D Graphics Environment. Computing in Science & Engineering, 9(3), 90–95.

[7] Django Software Foundation. Django Web Framework. https://www.djangoproject.com

[8] Flask Documentation. Flask: Web Development, One Drop at a Time. https://flask.palletsprojects.com

[9] GitHub Project Repository: fore_lood - Disaster Alert System using ML (2025). https://github.com/anantpandey01/fore_lood

[10] Hossain, M. S., et al. (2019). Early Warning System for Disaster Management using IoT and Machine Learning. IEEE Access, 7, 145742-145754.

[11] Tufekci, Z. (2014). Big Questions for Social Media Big Data: Representativeness, Validity and Other Methodological Pitfalls. ICWSM, 14, 505-514.

[12] Kaur, H., & Goyal, P. (2020). Natural Disaster Prediction using Ensemble Machine Learning Models. Procedia Computer Science, 173, 402–409.

[13] Liu, C., et al. (2021). Disaster Detection and Alerting System using AI and Cloud Computing. Journal of Cloud Computing, 10(1), 1-15.

[14] Raschka, S. (2015). Python Machine Learning. Packt Publishing Ltd