

## IoT Infrastructure For Disaster Anticipation & Response Using Machine Learning

Parth N. Panjabi<sup>1</sup>, Meher S. Gadge<sup>2</sup>, Om D. Mohiture<sup>3</sup>, Nidhi G. Jadhal<sup>4</sup>,  
Piyush S Bhende<sup>5</sup>

<sup>1,2,3,4,5</sup> B.E in Industrial Internet Of Things, Sant Gadge Baba Amravati University  
Amravati, Maharashtra, India

Guidance By: Prof. Rashmi A. Wakode

Received 08 April 2026; Accepted 20 April 2026

### ABSTRACT

Natural disasters strike without warning, often leaving communities vulnerable and unprepared. To address this challenge, our work explores how Artificial Intelligence (AI) and the Internet of Things (IoT) can be combined into a smart disaster anticipation system. Using sensors that track temperature, humidity, seismic activity, smoke, and water levels, the system collects real-time environmental data and processes it through machine learning models such as decision trees and neural networks. These models predict risk levels and deliver timely alerts through dashboards, SMS, and notifications, helping authorities and citizens respond faster. Security protocols ensure that data remains protected, while continuous learning allows the system to adapt to changing patterns over time. By integrating sensing, prediction, communication, and visualization, this framework demonstrates how AI-driven IoT solutions can strengthen disaster preparedness and reduce the impact of unpredictable natural events.

**KEYWORDS**—Artificial intelligence, IoT, Disaster Anticipation, Risk Prediction, Edge Processing, ML model, Cloud Dashboard, Real Time Alerts.

### I. INTRODUCTION

Natural calamities don't wait. They come suddenly, tearing through lives, trembling foundations, and flooding streets. Furthermore, every second counts during those crucial times. This project was motivated by the need to create a system that not only responds to but also foresees disasters. We envisioned a future in which technology pays attention to its surroundings—sensors that silently monitor trembling ground, rising water, or poisonous gases—and promptly warn people before danger materializes. We have developed a real-time disaster prediction system that tracks five major hazards: floods, earthquakes, landslides, cyclones, and volcanic eruptions. It uses IoT sensors and machine learning. At the core of our system is the ESP32 microcontroller, which connects to a set of disaster-specific sensors. These sensors provide real-time data to the Blynk Cloud, where it appears on a mobile dashboard and is processed by smart models. If something exceeds a danger level, the system reacts by sending alerts through sirens, notifications, or messages. This goes beyond just hardware and software. It focuses on people. It aims to help communities prepare, act, and stay safe. Whether it's a farmer in a flood-prone area or a rescue team in a landslide zone, our goal is clear: to turn raw data into life-saving choices. This project marks our move toward a safer, smarter future, where preparedness replaces fear and technology acts as a steady protector against nature's wrath.

### II. LITERATURE SURVEY

1] Amatya, S., Shreshtha, R. Shakya, S. et al. [2022]

This report explores the integration of IoT, ML, and edge computing in disaster management systems. The authors emphasize how predictive analytics can identify critical risk factors and forecast disaster trajectories using geospatial data. Their framework supports real-time notifications for events like floods and earthquakes, enhancing emergency response and resource coordination. Challenges such as data privacy, cost, and interoperability are also addressed, calling for scalable and ethical solutions in disaster resilience. Additionally, the study highlights the importance of rehabilitation services post-disaster, especially in low-resource regions. It advocates for scalable, ethical, and inclusive technology deployment.

2] *Satishkumar, D., & Sivaraja, M. [2024]*

In their book *Internet of Things and AI for Natural Disaster Management and Prediction*, the authors present a comprehensive academic framework for applying AI and IoT in disaster forecasting. They discuss practical implementations across floods, earthquakes, and cyclones, highlighting the role of embedded systems, sensor networks, and machine learning algorithms. The book bridges theory and practice, offering insights for policymakers and technologists aiming to safeguard lives through proactive disaster mitigation. The book also emphasizes the role of real-time data acquisition and cloud integration in improving situational awareness. Case studies from India and Southeast Asia are used to validate the proposed models

3] *Swapnil Bande & Dr. Virendra V. Shete [2017]*

This paper proposes a flood prediction model using environmental parameters like humidity, temperature, and rainfall. The authors employ Artificial Neural Networks (ANN) with the Levenberg- Marquardt algorithm, achieving ~88% accuracy. Their system demonstrates how sensor arrays and ML can be used for real-time flood forecasting, offering a cost-effective and scalable solution for urban and rural flood-prone zones. Their approach demonstrates how low-cost sensors and ML can be combined for scalable flood forecasting. The system is designed for both urban and rural deployment, with minimal infrastructure requirements.

4] **Jayashree, S., Sarika, S., Solai A. L., Soma Prathibha, Ramesh K., & Meenakshi V. [2017]** This paper presents a ZigBee-based flood alert system designed to function independently of mobile networks, making it ideal for emergency communication in remote or disaster-affected areas. The system monitors dam water levels and maps safe zones, allowing users to receive timely alerts. It includes features like SMS-based notifications, emergency call support, and user registration, ensuring personalized and accessible warnings.

5] *Kartika, Ni Komang Ega, Muhammad Ary, Murti, Casi Setianingsih [2019]*

This study presents an Android-based flood prediction app using Radial Basis Function Neural Networks and Antares API. The app forecasts flood risk for the next three months based on historical data. It showcases the potential of mobile platforms in democratizing disaster awareness and enabling early community-level preparedness. The authors highlight the importance of mobile accessibility in disaster awareness. Their model is trained on regional datasets, making it highly relevant for localized prediction.

6] *Wang J., Wang S., Zhang Y., et al. [2023]*

This paper investigates AI-driven disaster prediction systems using deep learning models for seismic, hydrological, and meteorological events. The authors compare neural networks with traditional statistical models, finding superior accuracy and responsiveness in AI-based systems. They also discuss limitations such as data quality, computational load, and ethical concerns, proposing architectural improvements for real-time deployment. They also explore the use of text-to-speech and navigation aids for post-disaster rehabilitation. Limitations such as small datasets and ethical concerns are discussed in depth.

7] **Sandarwan M. G. T., Madushanka M. K. P., & Hettige, Budditha., et al. [2025]**

Following PRISMA guidelines, this research reviews assistive technologies for disaster navigation. It highlights emerging IoT and ML solutions that enhance mobility and awareness during emergencies. The authors conducted a comprehensive literature search across IEEE Xplore, Google Scholar, and MDPI, identifying trends in wearable sensors, alert systems, and autonomous navigation aids for disaster-prone populations. The authors categorize technologies based on functionality navigation, alerting, and sensing. Their review also identifies gaps in affordability and adaptability across diverse terrains.

8] **Mr. Sanjeev Kumar, Avantika Srivastava, & Prajapati Kashish., et al. [2025]**

This paper introduces an AI-powered wearable system designed for real-time environmental sensing and alerting. It integrates object detection, scene understanding, and facial recognition to support individuals in high-risk zones. The system is lightweight, power-efficient, and optimized for wearable deployment, offering dynamic feedback and situational awareness without reliance on external infrastructure. The system is optimized for low power consumption and portability. It aims to bridge accessibility gaps in disaster-prone regions with minimal infrastructure.

9] **V. Chamola, V. Hassija, V. Gupta, & M. Guizani, et al. [2021]**

This study reviews the evolution of AI and ML in disaster prediction, starting with earthquake models and expanding to hurricanes, floods, and wildfires. The authors highlight the use of satellite data, vulnerability assessments, and real-time detection systems. Their bibliometric analysis reveals growing collaboration and

innovation in AI-powered disaster management. They propose a layered architecture combining edge computing and cloud analytics. The paper also discusses cybersecurity risks in IoT-based disaster systems.

10] *Martinez-Amaya, C., et al. [2023]*

Focusing on hurricane prediction, this paper presents a hybrid ML model combining satellite imagery and meteorological data. The system achieves high accuracy in forecasting storm paths and intensities. The authors emphasize the importance of real-time processing and cross-agency data sharing for effective early warning systems. The authors emphasize the need for cross-agency data sharing and real-time processing. Their model is validated using historical hurricane datasets from the Atlantic region.

### **III. PROBLEM STATEMENT**

#### **3.1 AIM**

To design and implement a real-time disaster anticipation system using IoT sensors and edge-integrated machine learning, capable of monitoring environmental parameters, predicting risk levels, and triggering timely alerts. The system aims to operate independently of external infrastructure, offering a low-cost, scalable solution for vulnerable communities. By combining sensor data acquisition, cloud visualization, and intelligent analysis, the project seeks to empower early response and enhance public safety in disaster-prone regions. Using edge-integrated machine learning and Internet of Things sensors, a real-time disaster anticipation system that can monitor environmental parameters, forecast risk levels, and send out timely alerts is to be designed and put into play. The system seeks to provide vulnerable communities with an affordable, scalable solution by functioning independently of external infrastructure. The goal of the project is to improve public safety and enable early response in disaster-prone areas by fusing sensor data collection, cloud visualization, and intelligent analysis.

#### **3.2 OBJECTIVE**

1. To use an ESP32 microcontroller and Internet of Things sensors to design and create a real-time disaster prediction system
2. To put machine learning algorithms into practice for disaster risk prediction and environmental data classification.
3. To incorporate alert systems and cloud-based visualization through the Blynk platform for remote monitoring.
4. To use sensors that are specific to disasters (such as temperature, vibration, gas, rain, and water level) to detect multiple hazards.
5. To develop a dashboard that is mobile-friendly and responsive, allowing for geo-mapped risk zones and immediate alerts.
6. To guarantee that the system is affordable, expandable, and prepared for field deployment in communities that are at risk.

### **IV. PROPOSED METHODOLOGY**

#### **5.1 PROPOSED SYSTEM**

Real-time environmental monitoring and risk prediction are made possible by the integration of a suite of disaster-specific sensors with an ESP32 microcontroller at the heart of the suggested system. These sensors, which are chosen to target particular natural hazards like floods, earthquakes, volcanic activity, and landslides, include temperature/humidity sensors, vibration modules, gas sensors, rain sensors, and water level detectors. Continuously gathered sensor data is sent over Wi-Fi to the Blynk Cloud, where it is displayed on a dashboard that can be viewed on a smartphone. Incoming data is processed by a machine learning model that was trained using decision tree algorithms. This model classifies risk levels and sends out alerts when thresholds are surpassed.

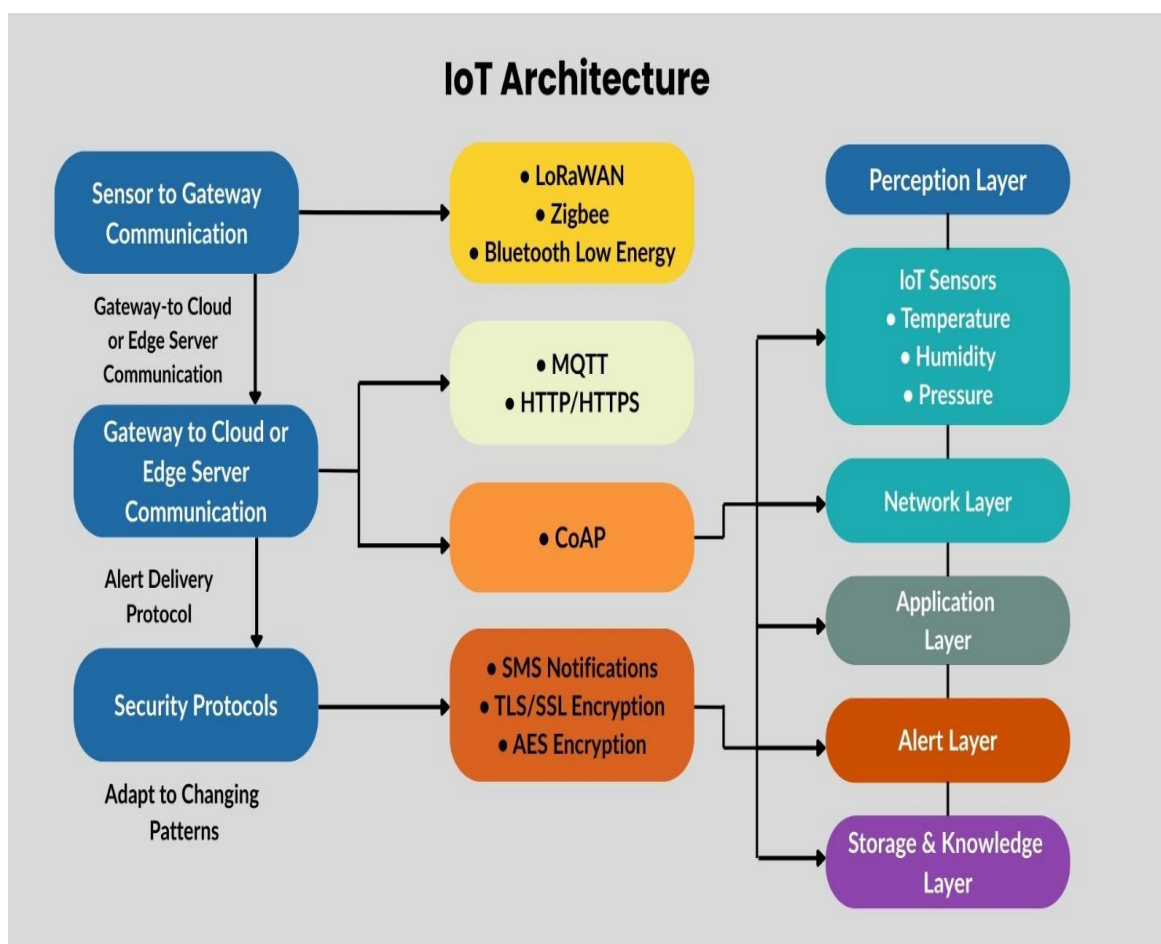
The system functions independently without the need for external infrastructure and is field-deployable, small, and inexpensive. Depending on local disaster profiles, its modular design makes expansion and customization simple. It is an effective and scalable solution for disaster anticipation at the community level because it combines cloud visualization, mobile access, and intelligent analysis.

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and, conversely, if there are not at least two sub-topics, then no subheads should be introduced. Styles named "Heading 1," "Heading 2," "Heading 3," and "Heading 4" are prescribed.

**5.2 Why We Use Edge ML**

- Fast response without internet dependency
- Reduced false alarms
- Pattern recognition instead of fixed threshold
- Offline intelligent decision making
- Energy efficient processing

**V. HOW COMMUNICATION PROTOCOL WORKS ?**



**Fig : 6.1 Architecture of Communication Protocol**

**6.1 DESCRIPTION OF ARCHITECTURE**

**Sensor-to-Gateway Communication**

These protocols connect field sensors to local gateways.

- LoRaWAN offers long-range, low-power transmission ideal for remote areas.
- Zigbee supports short-range mesh networking for dense sensor clusters.
- Bluetooth Low Energy is used for mobile or wearable sensors.
- Wi-Fi provides high-bandwidth connectivity in urban zones.

**Gateway-to-Cloud or Edge Server Communication**

This layer transmits aggregated sensor data to Processing units.

- MQTT is a lightweight protocol perfect for real-time IoT streaming.
- HTTP/HTTPS handles dashboard updates and API interactions.
- CoAP is optimized for constrained devices and intermittent networks.

#### Alert Delivery Protocols

These protocols ensure timely delivery of disaster Warnings.

- SMS reaches users even without internet access.
- Push Notifications alert mobile users instantly.
- Email/Webhooks integrate with emergency systems and dashboards.

#### Security Protocols

These protocols protect data and ensure secure access.

- TLS/SSL Encryption secures data in transit.
- OAuth 2.0 / JWT manages user authentication and access control.
- AES Encryption protects stored data on edge or cloud systems.

#### Perception Layer

This layer includes IoT sensors that monitor Environmental parameters.

- Sensors track temperature, humidity, seismic activity, smoke, and water level.
- Data is collected continuously and sent to gateways for processing.
- It acts as the “eyes and ears” of the system.
- Sensor accuracy and placement are critical for early detection.

#### Network Layer

This layer handles data transmission from sensors to Processing units.

- Gateways aggregate sensor data and forward it using MQTT or CoAP.
- Edge nodes may perform basic filtering or buffering.
- Reliable communication ensures minimal data loss.

#### Processing Layer

This is the intelligence core of the system.

- Preprocessing cleans and normalizes incoming data.
- ML models (Decision Tree, Neural Network) classify risk levels.
- Security protocols protect sensitive data during analysis.
- It transforms raw data into actionable insights.

#### Application Layer

This layer presents insights to users and authorities.

- Dashboards display real-time maps, graphs, and alerts.
- Interfaces are accessible via mobile and web platforms.
- It simplifies complex analytics for decision-makers.
- User experience is optimized for clarity and speed.

#### Alert Layer

This layer activates warnings based on predictions.

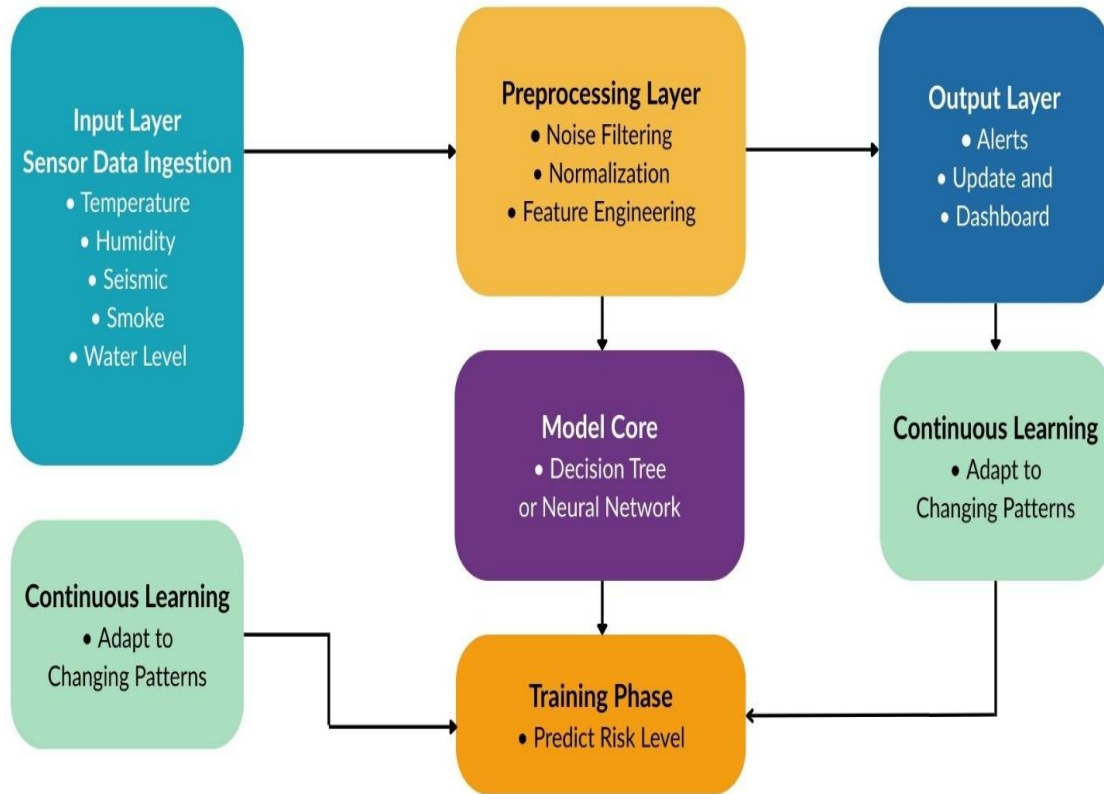
- Alerts are sent via SMS, push notifications, and buzzers.
- Multi-channel delivery ensures redundancy and reach.
- Timing is critical — alerts must be near-instant.
- It’s the system’s voice during emergencies.

#### Storage & Knowledge Layer

This layer stores historical data and model learnings.

- Cloud databases archive sensor readings and predictions.
- Knowledge bases help retrain models over time.
- Enables continuous improvement and adaptation.
- It’s the memory and learning engine of the system.

**VI. HOW M.L MODEL WORKS**



**Fig 7.1 : Architecture of Machine Learning Model**

**7.1 DESCRIPTION OF ML MODEL ARCHITECTURE**

**1. Input Layer**

This is the entry point where sensor data is collected. Sensors measure temperature, humidity, seismic activity, smoke, and water level continuously. Each reading becomes a structured feature for the ML pipeline. The diversity of inputs ensures detection of multiple disaster types. Accurate sensors reduce false alarms and improve reliability. Data is streamed in real time for immediate processing. It acts as the “eyes and ears” of the system. Without this layer, the model has no foundation to work on.

**2. Preprocessing Layer**

Raw sensor data often contains noise and inconsistencies. Noise Filtering removes faulty or extreme values. Normalization scales readings into a uniform range. Feature Engineering derives new variables like rate of change.

This step ensures the model receives clean, meaningful data.  
It reduces bias and improves model accuracy.  
Preprocessing also handles missing or incomplete data.  
It's the "cleaning and shaping" stage before intelligence.

### 3. Model Core

This is the intelligence engine of the system.  
It can use a Decision Tree (rule-based, interpretable).  
Or a Neural Network (pattern recognition, complex learning).  
The model learns correlations between sensor features and disasters.  
It outputs classifications such as low, medium, or high risk.  
Decision Trees are fast and explainable for edge devices.  
Neural Networks capture deeper, non-linear patterns.  
This stage is the "brain" where predictions are generated.

### 4. Training Phase

The model is trained using historical sensor data and disaster records.  
It learns which patterns precede floods, earthquakes, or fires.  
Loss functions measure prediction errors during training.  
Weights are adjusted to minimize error (backpropagation in neural nets).  
Training requires large, diverse datasets for robustness.  
It builds the knowledge base for future inference.  
Overfitting is avoided by validation techniques.  
This phase is the "learning memory" of the system.

### 5. Inference Phase

Here, the trained model is applied to new real-time data.  
It predicts the risk level (e.g., "High flood risk").  
Outputs include confidence scores (e.g., 92% probability).  
Predictions are fast, continuous, and adaptive.  
It ensures communities receive timely warnings.  
Inference is lightweight for deployment on edge devices.  
It's the system's real-time decision-making engine.  
Accuracy here determines trust in the system.

### 6. Output Layer

The results are communicated to users and authorities.  
Alerts are sent via SMS, push notifications, buzzers, or dashboards.  
Dashboards visualize data with maps, graphs, and indicators.  
This layer translates analytics into actionable insights.  
It ensures clarity for both technical and non-technical users.  
Alerts are multi-channel for redundancy.  
It acts as the "voice" of the system during emergencies.  
Timely output saves lives and resources.

### 7. Continuous Learning

The system adapts to changing patterns over time.  
New sensor data and disaster outcomes are stored.  
Models are retrained periodically to improve accuracy.  
This ensures resilience against seasonal or environmental shifts.  
It prevents the system from becoming outdated.  
Feedback loops refine predictions continuously.  
It's the "memory and adaptability" of the system.  
Without this, the model would stagnate and lose relevance.

## VII. SYSTEM REQUIREMENTS

### a. *HARDWARE REQUIREMENT*

- i. Arduino Uno Microcontroller
- ii. Water Flow Sensor
- iii. Vibration Sensor (ADXL 345)
- iv. Gas Sensor
- v. Fire Sensor
- vi. Temperature & Humidity Sensor (DHT11)
- vii. Breadboard and Jumper Wires (for prototyping)
- viii. Power Supply Module or Rechargeable Battery Pack
- ix. Buzzer / Siren Module (for alert activation)
- x. LED Display (16\*2)

### b. *SOFTWARE REQUIREMENT*

- i. Firmware: Arduino IDE (for ESP32 programming)
- ii. Programming Language: Python
- iii. ML Model: Decision Tree (trained using Python)

### c. *MODULE USED*

- i. ESP32 Microcontroller – Collects and transmits sensor data in real time.
- ii. Disaster-Specific Sensors – Detect environmental changes (Water, Vibration etc).
- iii. Machine Learning Model – Classifies sensor inputs to predict disaster risk.
- iv. Alert Module (Buzzer)– Delivers real-time warnings during emergencies.

## VIII. ADVANTAGES

### 1]. **Real-time audio guidance**

The system acts like a companion, offering continuous audio cues that help users stay aware of their surroundings.

### 2]. **Independence and mobility**

By reducing reliance on external assistance, the solution empowers visually impaired individuals to move with greater confidence.

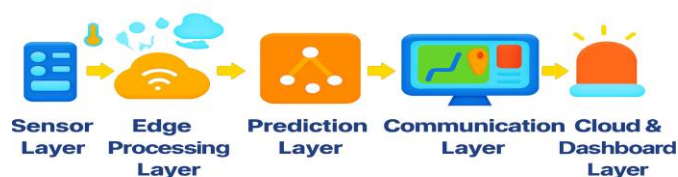
### 3]. **Compact and wearable design**

Designed to be lightweight and unobtrusive, the device fits seamlessly into daily life.

### 4]. **Affordable and scalable solution**

Unlike many traditional assistive technologies that are expensive and limited in reach, this system is cost-effective and adaptable.

## X. BLOCK DIAGRAM OF MODEL



### Layer-by-Layer Breakdown

1. **Sensor Layer**
  - Collects real-time data from sensors: temperature, humidity, seismic, smoke, water level.
  - Acts as the foundation of the system's awareness.

- 2. **Edge Processing Layer**
  - Performs local filtering, normalization, and feature extraction.
  - Reduces latency and bandwidth before cloud transmission.
- 3. **Prediction Layer**
  - Uses ML models (Decision Tree, Neural Network) to classify risk levels.
  - Core intelligence engine for disaster anticipation.
- 4. **Communication Layer**
  - Transmits predictions to users via dashboards, alerts, and notifications.
  - Ensures timely and clear delivery of insights.
- 5. **Cloud & Dashboard Layer**
  - Stores historical data and visualizes real-time analytics.
  - Supports continuous learning and centralized monitoring.

### XI. DISADVANTAGES

- **Data Quality Issues:** Poor or missing sensor data can lead to false alarms or missed disasters.
- **High Cost:** Deploying sensors, gateways, and cloud infrastructure is expensive to set up and maintain.
- **Latency Risks:** Heavy models or weak networks may delay alerts, reducing effectiveness.
- **Security Concerns:** Vulnerable to hacking or data breaches if encryption and access control are weak.

### XII. LIMITATIONS

- **Regional Adaptability:** Models trained in one area may not perform well in another due to different disaster patterns.
- **Infrastructure Dependency:** Requires stable power and connectivity, which may fail during disasters.
- **Continuous Retraining Needed:** Models must be updated regularly to adapt to new data and changing conditions.
- **Trust & Transparency:** Communities may distrust automated alerts without clear explanations or human oversight.

### XIII. RESULTS

#### 13.1 Visual Analysis of Outcomes

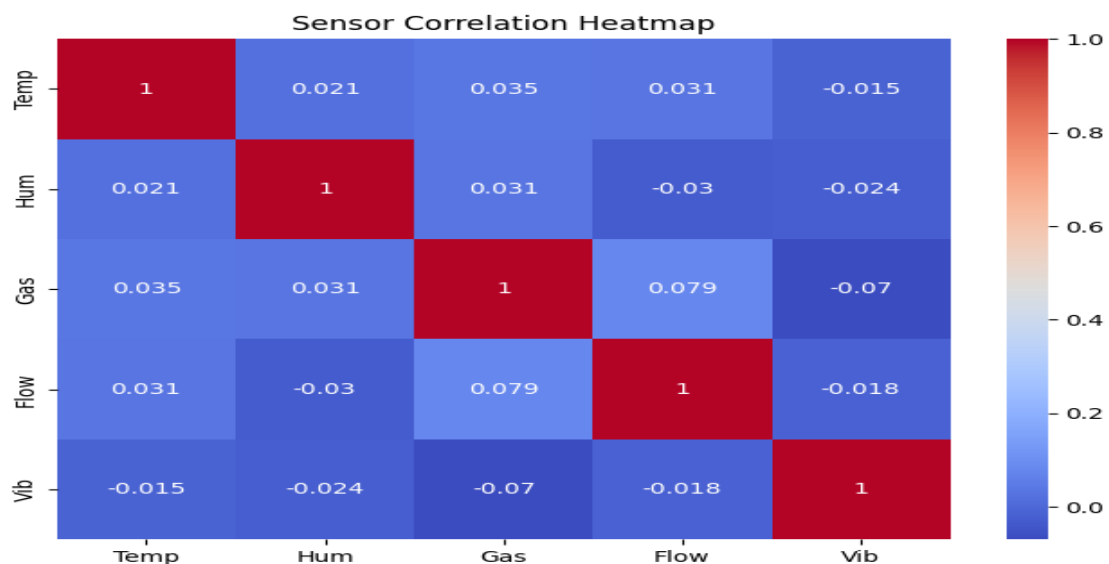


Fig 13.1 : Sensor Correlation Heatmap

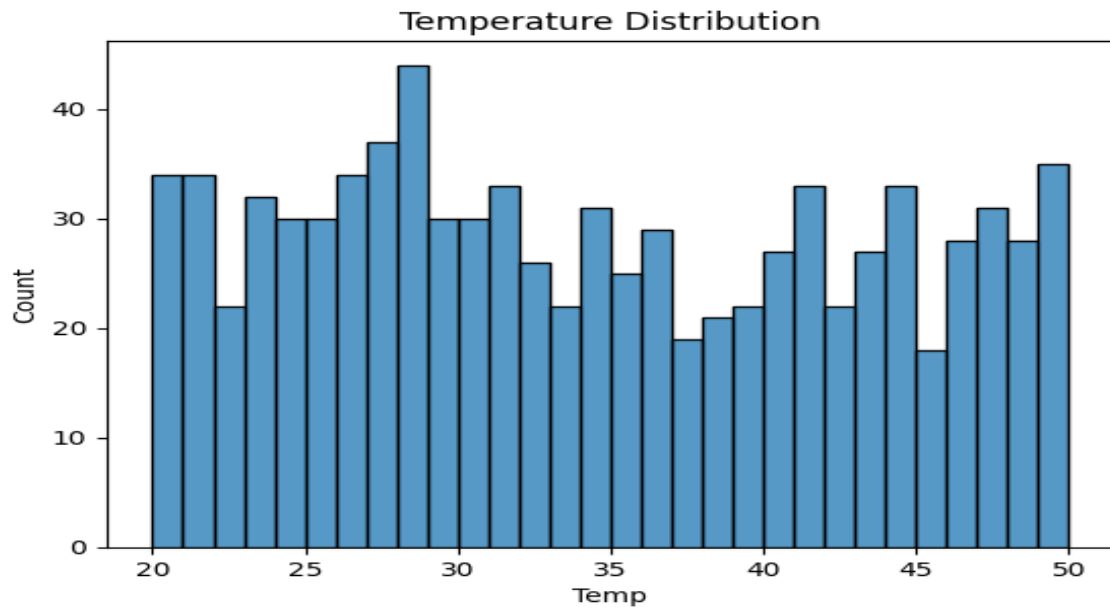


Fig 13. 2: Temperature Distribution

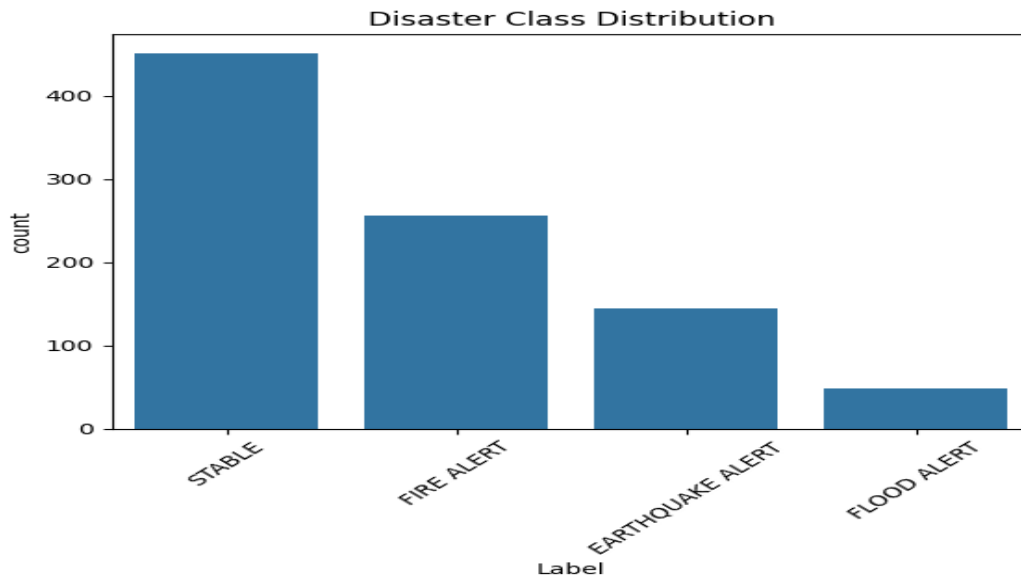


Fig 13.3 : Disaster Class Distribution

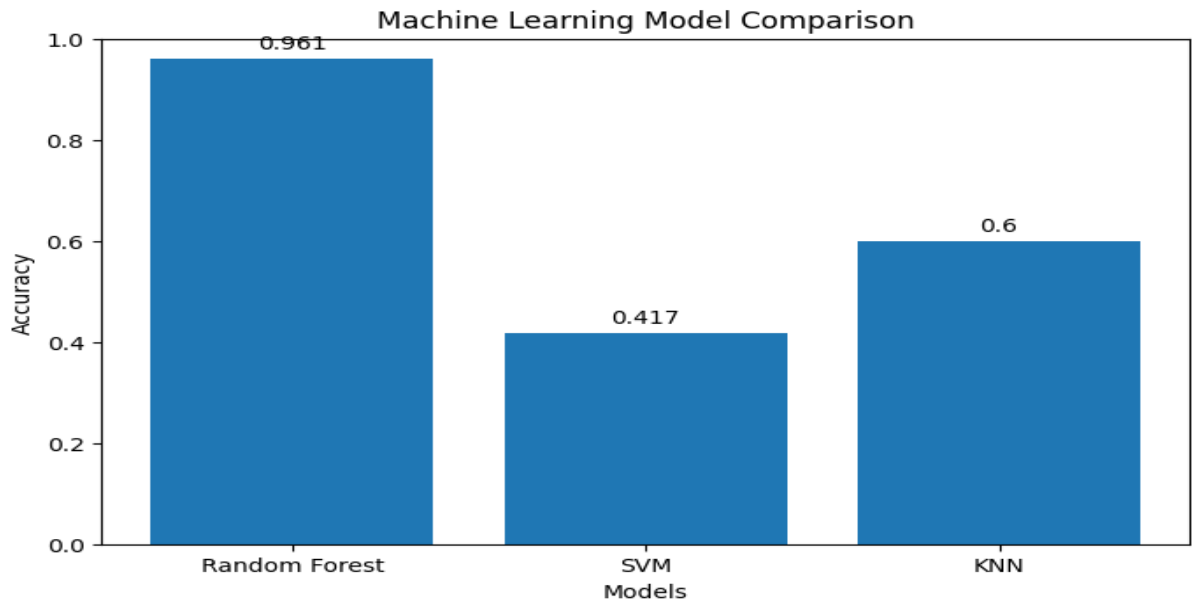


Fig 13.4 : Machine Learning Model Comparison

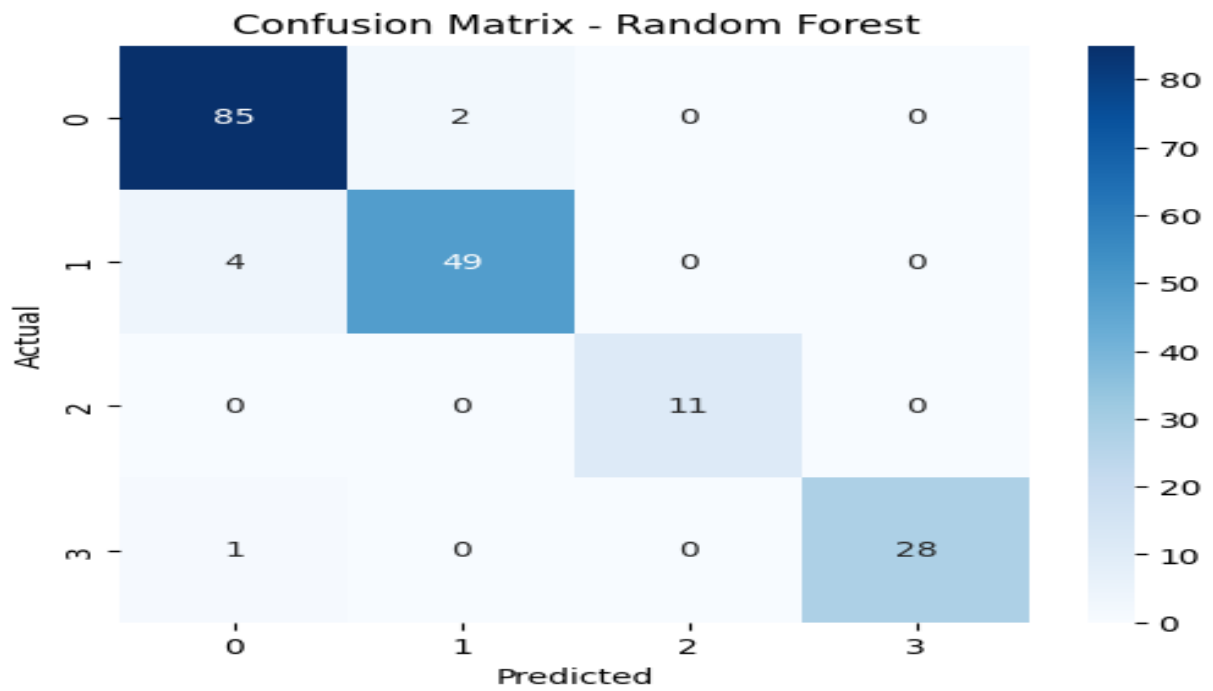


Fig 13.5 Confusion Matrix

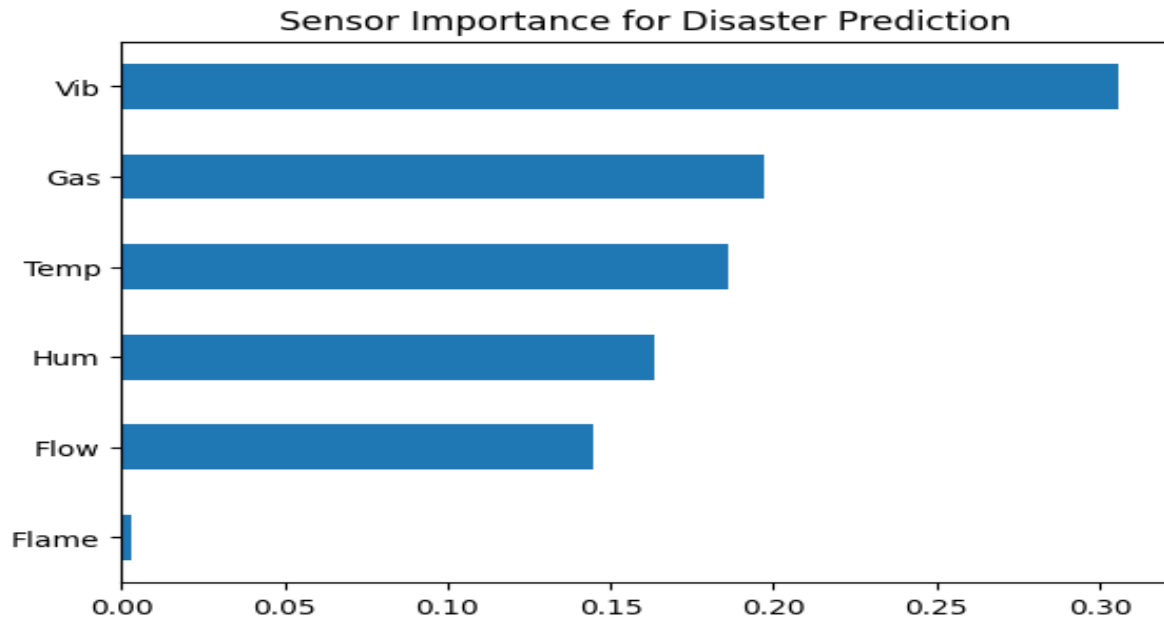
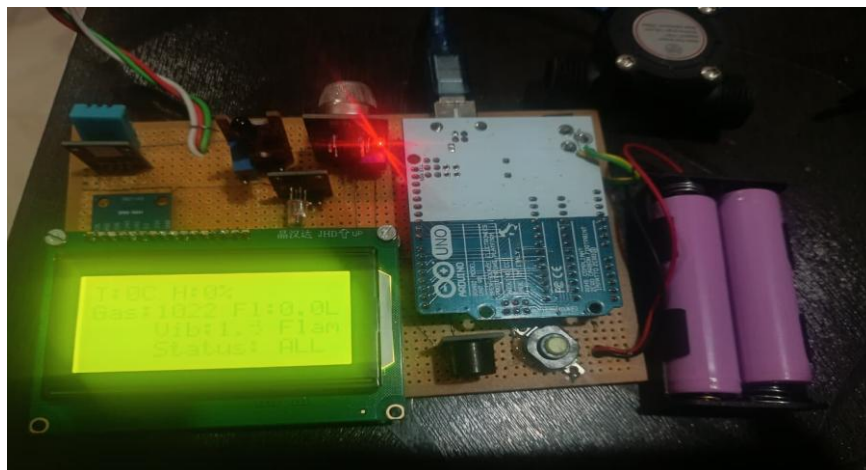
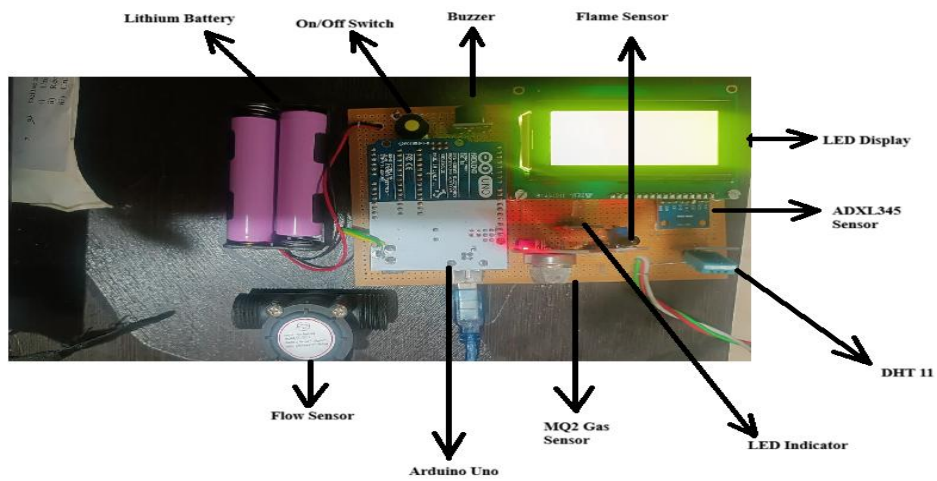


Fig 13.6 Sensor Importance

### 13.2 Model Images



#### XIV. CONCLUSION

Our proposed system does more than predict disasters — it listens to the environment, learns from it, and speaks up when danger approaches. By combining IoT sensors with machine learning, we have created a framework that can detect early signs of floods, earthquakes, landslides, and other hazards, giving communities the chance to act before it is too late.

This project is not just about hardware and algorithms; it is about **protecting lives and empowering people**. With real-time alerts, mobile dashboards, and a compact design, the system is built to serve even in remote or vulnerable areas. Unlike traditional infrastructures that demand heavy resources, our solution is light, smart, and scalable — designed to reach those who need it most.

By turning raw sensor data into **meaningful decisions**, the system becomes a quiet guardian against nature's unpredictability. It offers more than awareness; it provides a sense of control, safety, and resilience. In doing so, it helps communities prepare, respond, and recover with confidence, showing how innovation can stand as a shield in the face of uncertainty.

#### REFERENCES

- [1]. Amatya, S., Shrestha, R., & Shakya, S. "Rehabilitation-focused disaster management using IoT and AI." *International Journal of Disaster Risk Reduction* 70 (2022): 102789.
- [2]. Satishkumar, D., & Sivaraja, M. *Internet of Things and AI for Natural Disaster Management and Prediction*. CRC Press, 2024.
- [3]. Swapnil Bande, Virendra V. Shete. "Flood prediction using ANN and environmental parameters." *International Journal of Engineering Research & Technology (IJERT)* 6, no. 6 1–5.
- [4]. Jayashree, S., Sarika, S., Solai, A. L., Soma Prathibha, Ramesh, K., & Meenakshi, V.. "ZigBee- based flood alert system for network-independent emergency communication." *International Journal of Engineering and Technology* 9, no. 2 (2017): 120–125.
- [5]. Kartika, Ni Komang Ega, Murti, Muhammad Ary, & Setianingsih, Casi. "Flood prediction using RBF Neural Networks and Antares API." *Journal of Physics: Conference Series* 1231 (2019): 012012.
- [6]. Wang J. , Wang S. , Zhang Y. "Artificial intelligence for disaster prediction and response." *Displays* 77 (2023): 102391.
- [7]. Sandaruwan M. G. T., Madushanka M. K. P., & Hettige , Budditha. "Artificial intelligence- based solution for navigating disaster-prone environments: A review." *Machine Learning* 7 (2025): 8.
- [8]. Mr. Sanjeev Kumar, Avantika Srivastava, & Prajapati Kashish "AI-powered wearable system for environmental sensing." *International Journal of Trendy Research in Engineering and Technology* 9, no. 4 (2025): 45–52.
- [9]. Chamola, V., Hassija, V., Gupta, V., & Guizani, M. "Disaster management using IoT and machine learning." *IEEE Internet of Things Journal* 8, no. 21 (2021): 16200–16217.
- [10]. Martinez-Amaya, C., et al. "Hybrid ML model for hurricane prediction using satellite data." *Remote Sensing* 15, no. 3 (2023): 789–804.
- [11]. Dodangeh, M., et al. "Edge AI for disaster response: A low-latency framework." *Sensors* 20, no. 18 (2020): 5123.