**DISASTER RECOGNITION AND MANAGEMENT USING API**

**Roll Number Student Name 20211ISR0078 Bhavana B A**

**2011ISR0021 Monika P**

**20211ISR0038 Disha R Under The Supervision of**

**Dr. Sivaramakrishnan.S Associate Professor**

**School of Computer Science Engineering & Information Science**

**Presidency University**

## Introduction

Natural and man-made disasters pose risks to life, infrastructure, and the environment. Traditional information systems rely on official feeds and databases, which are often delayed, non-localized, or inaccessible. In today's data-rich world, there's a growing need for intelligent systems that can extract real-time, structured insights from unstructured sources.

To address this, a **real-time Disaster Information Aggregation Dashboard** has been developed. The system leverages **OpenAI’s GPT-4** and **zero-shot learning** to understand natural language queries and provide up-to-date disaster information without task-specific training. Prompt engineering is used to extract meaningful insights from live, unstructured web data.

The backend is built with **Node.js and Express.js**, while the frontend uses **React.js** for an intuitive user interface. Visualizations are rendered using **Leaflet.js** for maps and **Chart.js** for dynamic charts. Unlike traditional systems, this dashboard fetches data live per request and does not rely on a persistent database.

The platform uses **Natural Language Understanding (NLU)** to extract key disaster attributes—type, location, severity, and sentiment—making it highly suitable for emergency response, travel safety, and policymaking.

### **Objectives of this Project:**

* **To develop a smart dashboard** that aggregates real-time disaster data from unstructured sources using AI.
* **To eliminate reliance on static databases** by integrating real-time web-based querying.
* **To enable natural language interaction** through GPT-4 and zero-shot learning.
* **To visualize disaster data** effectively using interactive maps and charts.
* **To support better decision-making** in emergency response, travel, and policy formulation.

## Literature Review

Over the past decade, **AI, NLP, and geospatial technologies** have transformed disaster management by enabling automation, precision, and real-time situational awareness. Several foundational studies guided the development and architecture of this dashboard.

**Patel et al.** proposed an AI-based earthquake response system using deep learning and BERT to classify real-time seismographic data and disaster-related text, stressing the importance of automated classification in fast-evolving scenarios.

**Kumbam and Vejre’s FloodLense** utilized ChatGPT to extract insights from real-time flood reports, showing how transformer models can analyze unstructured text to generate relevant disaster information.

**Tripathy et al.** introduced a secure cloud-IoT model emphasizing access control and safe data transmission. While their focus was on security, it influenced this project’s emphasis on scalable, lightweight data flows without permanent storage.

**Saha et al.** leveraged IoT and cloud systems for real-time disaster monitoring using sensors. In contrast, our system replaces physical sensors with AI-driven real-time web data processing.

**Mody et al.** developed Distress, a user-centric emergency alert app. Their design principles inspired our intuitive React-based interface and real-time map visualization using Leaflet.js.

**Xu’s research** on sentiment analysis using VADER showcased how public sentiment reflects disaster severity. This technique was integrated into our system to add emotional context to disaster reports.

**Rajasekharan et al.** emphasized structured databases for disaster data, whereas our approach avoids persistent databases in favor of fresh, real-time data using GPT-4 for immediate relevance.

**Rajabifard et al.** highlighted spatial data infrastructures for disaster visualization, influencing our location-based mapping using geospatial coordinates in Leaflet.js.

**Wallace and De Balogh** underlined the role of real-time dashboards in accelerating emergency response, which guided our decision to build a dynamic, interactive interface for faster decision-making.

**Kalidoss and Ravi** presented models using IoT and big data for low-latency disaster response. Their vision aligns with our GPT-4-powered system designed for live data processing and rapid visualization.

Together, these studies shaped a scalable, AI-driven, and user-friendly disaster dashboard capable of real-time understanding, visualization, and response using the latest in NLP and web technologies.

## Research Gaps Identified

* Despite major strides in AI, NLP, and geospatial analytics, existing disaster information systems continue to face critical limitations.
* One of the key challenges is their reliance on **predefined and labeled datasets**, which reduces the adaptability of AI models. Such models often require task-specific training, limiting their effectiveness when confronted with unfamiliar disaster types or varied user queries. This rigidity prevents the system from responding flexibly to evolving real-world scenarios.
* Another significant limitation arises from the **hardware dependency of IoT-based systems**. While they contribute valuable early warnings, these systems are impractical for global deployment—especially in low-resource or rural areas lacking the necessary infrastructure. This makes such systems inaccessible in regions where disaster response tools are needed the most.
* Furthermore, most current platforms are not capable of handling **free-form natural language queries**, making them difficult to use for non-technical users. They often separate data visualization and sentiment analysis into isolated modules, failing to offer an integrated, holistic view of a disaster event—its **type, severity, public perception**, and **geographical spread**—all in one intuitive interface.

### ****How the Project Fills the Gap****

* The **Disaster Information Aggregation Dashboard** addresses these challenges with a real-time, infrastructure-light, and highly adaptive solution. The system is built atop **GPT-4** and leverages **zero-shot learning**, allowing it to interpret and respond to natural language queries without the need for predefined training or labeled data. This ensures **flexibility**, **adaptability**, and **robust performance** even in the face of emerging disaster types or unpredictable user queries.
* By **retrieving data live from the web**, the system eliminates the delays and maintenance overhead associated with static databases. This ensures that users are always presented with the most up-to-date information, which is critical during rapidly unfolding disasters.
* Unlike conventional tools, this platform **integrates sentiment analysis, statistical visualizations (Chart.js), and geospatial mapping (Leaflet.js)** into a **single cohesive interface**. This enables users to visualize the full impact of a disaster—including real-time public sentiment and geographic scope—at a glance.
* Additionally, the system is **browser-based** and requires **no physical infrastructure**, making it universally deployable and accessible even in regions with limited technological resources.
* Finally, by supporting **free-form, natural language queries**, the dashboard empowers users of all technical backgrounds to access and understand disaster data easily. Its intuitive design and NLP-powered interactions eliminate the need for specialized knowledge, making the system significantly more **inclusive and user-friendly** than existing platforms.

## Proposed Methodology

The Disaster Information Aggregation Dashboard employs a real-time, AI-driven architecture that integrates GPT-4, geospatial visualization, and sentiment analysis to deliver timely and comprehensive insights on disaster events. The methodology consists of the following key components:

#### **1. **Real-Time Data Fetching and Integration:**** Unlike traditional systems that rely on static datasets, this platform dynamically retrieves disaster data using APIs from reliable real-time sources such as news outlets, disaster management agencies, and public databases. The system collects information on disaster type, severity, location, and status, ensuring the data remains current without manual syncing or updates.

#### **2. **Natural Language Processing and Zero-Shot Learning**:**The system leverages GPT-4’s zero-shot learning capabilities to handle natural language queries without the need for prior dataset training. Users can ask free-form questions like “What’s the latest update on the earthquake in Nepal?” and receive accurate responses. This enables flexible interaction with the system, making it scalable and adaptable to new disasters or unconventional query phrasing.

#### **3. **Geospatial Visualization**:**Interactive maps powered by Leaflet.js allow users to view disaster locations through real-time geospatial markers. Users can zoom, pan, and click on affected areas to see detailed information. This enhances situational awareness and helps visualize the scale and geographic impact of events.

#### **4. **Sentiment Analysis;****The system incorporates sentiment analysis using tools like VADER to assess public response from real-time news and social media content. Sentiment scores are visualized using Chart.js, offering insights into public emotions—such as concern, panic, or hope—alongside factual data. This adds a social and emotional context to the disaster.

#### **5. **User Interface and Interaction Design:****Designed with user accessibility in mind, the dashboard supports interaction via natural language input, map navigation, and data visualizations. The web-based interface is fully responsive and device-independent, making it accessible from both high-resource urban areas and low-resource rural environments.

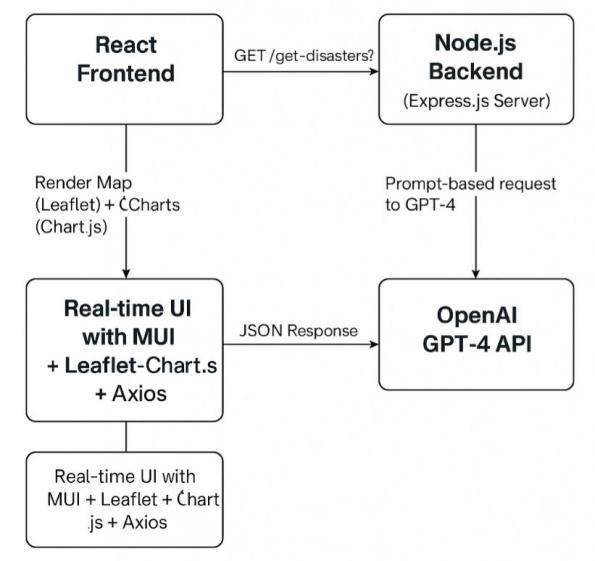
#### **6. **Real-Time Alerts and Notifications:****The platform supports live alerts for new or worsening disasters. Users can customize notifications based on location, disaster type, or severity. This ensures timely updates and helps users stay informed and ready to respond.

## Objectives

The Disaster Information Aggregation Dashboard aims to offer a real-time, AI-powered platform that enables users to access accurate disaster-related data with ease. The primary objectives include:

* **Deliver Real-Time Disaster Information:**The platform fetches live data on disaster types (e.g., floods, earthquakes), severity, and affected locations using authorized online sources, ensuring the most up-to-date insights.
* **Enable Natural Language Interaction:** Users can ask questions in plain language, such as “What’s the status of the flood in Assam?”, and receive relevant answers through GPT-4’s zero-shot learning capabilities—without predefined datasets or training.
* **Visualize Disasters with Interactive Mapping:** Leveraging Leaflet.js, the system provides a dynamic map highlighting disaster zones and severity levels, allowing users to visually understand geographic impact in real time.
* **Analyze Public Sentiment:** Using tools like VADER, the platform assesses real-time public sentiment from news and social media, offering insights into public emotions, concerns, and awareness around ongoing disasters.
* **Ensure Scalability and Flexibility:** Built on cloud infrastructure, the system supports high data volumes and user traffic while allowing easy integration of new sources and features, making it adaptable and future-ready.
* **Centralize All Disaster Information:** The dashboard consolidates key data—real-time updates, sentiment analysis, maps, and query responses—into one accessible interface, eliminating the need to visit multiple sources.
* **Personalized User Experience;** Users can customize notifications based on disaster type, location, and severity, ensuring timely and relevant updates tailored to their preferences.
* **Cross-Platform Accessibility:** Accessible via any internet-enabled device and supported by modern browsers, the platform ensures availability across urban and rural areas alike.
* **Support Informed Disaster Response:**By providing accurate and timely data, the platform aids government bodies, emergency services, and citizens in making faster, better-informed decisions during disasters.
* **Promote Awareness and Readiness:** The system fosters global awareness and preparedness by offering easy access to disaster updates, helping individuals and agencies plan and respond more effectively.

System Design And Implementation



### ****System Architecture Overview****

The Disaster Information Aggregation Dashboard follows a **three-tier architecture**, comprising:

#### ****1. Frontend (User Interface)****

Developed using **React.js**, the frontend allows users to interact with the system by submitting natural language queries (e.g., “Where is the nearest flood occurring?”).

* **Leaflet.js** powers an interactive disaster map showing affected regions and severity levels.
* **Chart.js** presents statistical data through visualizations (e.g., number of disasters, severity, affected areas).
* The interface is intuitive and delivers responses in real time, enhancing user experience.

#### ****2. Backend (Server & API Communication)****

The backend is built using **Node.js** and **Express.js**.

* It receives user queries from the frontend and structures them into prompts.
* These prompts are sent to **OpenAI’s GPT-4 API** for processing.
* Once a response is received, it’s formatted and returned to the frontend for display.This layer handles all communication between the frontend and the AI model, ensuring fast and reliable data processing.

#### ****3. GPT-4 (Query Processing AI Model)****

At the core of the system is **OpenAI's GPT-4**, which uses **zero-shot learning** to respond to disaster-related queries.

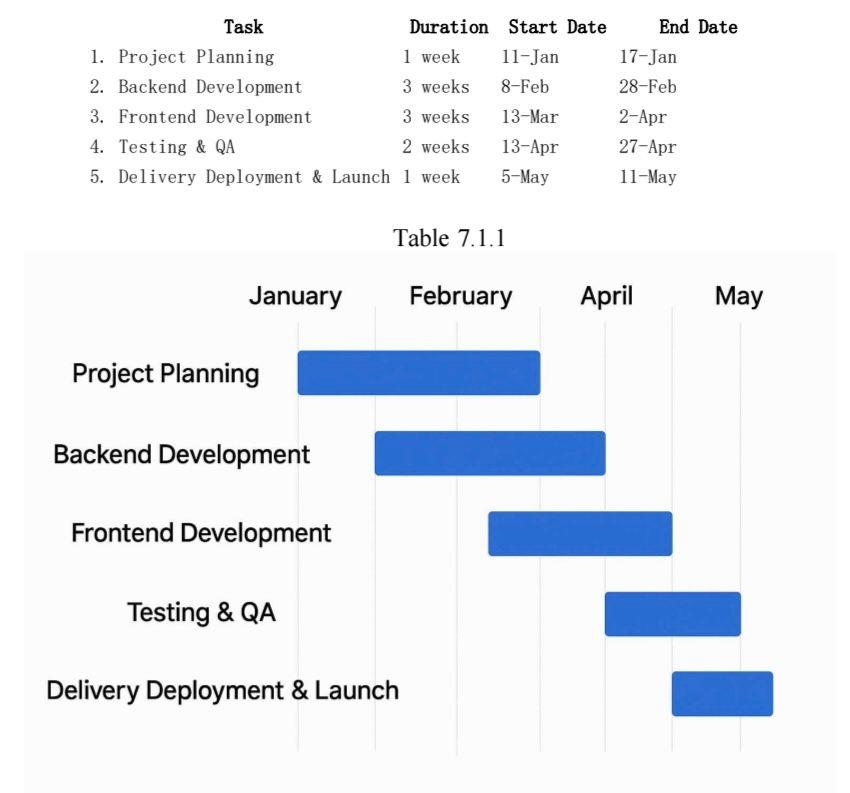
* It does not rely on predefined disaster-specific datasets.
* Instead, it utilizes general language understanding and inference to generate accurate and context-aware responses.
* This allows the system to adapt to a wide range of disaster-related questions without the need for specialized training.

#### ****4. Real-Time Data Updates****

To maintain live updates:

* The system uses **WebSockets** or **long polling** to establish a continuous connection between the backend and frontend.
* Users receive real-time updates on disaster data without refreshing the page.
* This ensures timely access to critical information during rapidly evolving disaster events.

Timeline of Project



# Outcomes / Results Obtained

1. The **Disaster Information Aggregation Dashboard** effectively handles real-time disaster queries using **GPT-4** combined with advanced **Natural Language Processing (NLP)**. Without relying on a conventional database, it identifies ongoing disasters, aggregates key data (severity, location, distribution), and presents it in an intuitive format. This ensures real-time accuracy and adaptability.
2. The system is built on two core techniques: **Zero-shot Learning** and **Prompt-based Querying**, both powered by GPT-4. Prompt-based querying involves framing specific prompts to extract detailed disaster information—such as nature, date, location, severity, and geolocation—from GPT-4 via a **Node.js and Express.js** backend.
3. Zero-shot learning allows GPT-4 to answer queries it hasn’t been explicitly trained on, thanks to its extensive pre-training. This gives the system flexibility in handling varied or unexpected user inputs.
4. GPT-4, based on the transformer architecture, uses **self-attention**, **positional encoding**, and **contextual word prediction** to generate coherent and accurate responses. Its strong **Natural Language Understanding** enables it to extract relevant disaster details even from unstructured text.
5. The user interface, built with **React.js** and **Leaflet.js**, displays disaster data on an interactive map using coordinates provided by GPT-4. **Chart.js** is used for visualizing data through bar and pie charts, while sentiment analysis adds emotional context based on disaster-related reports.
6. A key innovation is the system’s **real-time querying model**. It doesn’t store data; instead, each user query triggers a fresh API call to GPT-4, ensuring all information is current and relevant.

# Conclusion

* The Disaster Information Aggregation Dashboard offers a modern, efficient solution for real-time disaster tracking and information delivery. Leveraging cutting-edge technologies like OpenAI GPT-4, Prompt-based Querying, Zero-shot Learning, and Natural Language Understanding, it gathers disaster data from the internet without relying on traditional databases. This approach ensures speed, scalability, and up-to-date, contextually accurate information.
* The system features a React.js frontend and a Node.js + Express.js backend, providing a user-friendly interface. Visualizations using Leaflet.js and Chart.js present disaster data through maps, graphs, and sentiment analysis, enabling real-time dynamic data flow from queries to display.
* Secure protocols maintain data privacy while GPT-4’s adaptability through Zero-shot Learning allows the dashboard to handle new disasters and evolving queries without specific training.
* In summary, this project demonstrates how prompt engineering and AI can create scalable, intelligent, user-focused disaster information systems with strong potential for integration into global early warning and emergency management platforms.

# References

1. R. Kumbam and K. M. Vejre, “FloodLense: A Framework for ChatGPT-based Real-time Flood Detection,” arXiv, 2024.
2. D. Le, “Disaster Tweets Classification using BERT-Based Language Model,” arXiv, 2022
3. A. K. Tripathy, S. K. Sahu, and A. K. Turuk, "An efficient and secure data sharing scheme for cloud-based Internet of Things," IEEE Internet of Things Journal, vol. 8, no. 1, pp. 197–204, Jan. 2021.
4. H. N. Saha et al., "Disaster management using Internet of Things," Proc. 2017 8th Annu. Ind. Autom. Electromech. Eng. Conf. (IEMECON), Bangkok, 2017.
5. V. Mody et al., "Distress – An application for emergency response and disaster management," Proc. 2020 Int. Conf. Smart Electron. Commun. (ICOSEC), Trichy, 2020.
6. K. Chanda, “Efficacy of BERT embeddings on predicting disaster from Twitter data,” arXiv, 2021.
7. R. S. Rajasekharan et al., "A study on disaster management databases," Proc. 2021 Int. Conf. Military Technol. (ICMT), Brno, 2021.
8. Patel, P. Bhattacharjee, A. Reza, and P. Pradhan, “Earthquake Response Analysis with AI,” arXiv, 2025.
9. Bass, P. Clements, and R. Kazman, Software Architecture in Practice, 2nd ed., Pearson, 2003
10. T. J. Kalidoss and S. Ravi, "Disaster management system leveraging emerging digital technologies," Proc. 2016 IEEE Int. Conf. Comput. Intell. Comput. Res. (ICCIC), Chennai, 2016

# Publication Details

# pubb

Thank You