VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY
An Autonomous Institute Affiliated to University of Mumbai
Department of Computer Engineering

Project Report on

ResQconnect: AI-Driven Disaster Management System

In partial fulfillment of the Fourth Year, Bachelor of Engineering (B.E.) Degree in Computer Engineering at the University of Mumbai Academic Year 2024-25

Submitted by
Sai Thikekar (D17 - A , Roll no - 64)
Aradhya Ingle (D17 - A , Roll no - 24)
Arya Banavali (D17 - A , Roll no - 01)
Yash Chhaproo (D17 - A , Roll no - 07)

Project Mentor
Dr. Rohini Temkar

(2024-25)

VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY
An Autonomous Institute Affiliated to University of Mumbai
Department of Computer Engineering

Certificate

This is to certify that ***Sai Thikekar (D17-A, 64), Aradhya Ingle (D17-A, 24), Arya Banavali (D17-A, 01), Yash Chhaproo (D17-A, 07)*** of Fourth Year Computer Engineering studying under the University of Mumbai have satisfactorily completed the project on "***ResQconnect: AI Driven Disaster Management System***" as a part of their coursework of PROJECT-II for Semester-VIII under the guidance of their mentor ***Prof. Rohini Temkar*** in the year 2024-25.

This project report entitled ***ResQconnect: AI Driven Disaster Management System*** by ***Sai Thikekar, Aradhya Ingle, Arya Banavali, Yash Chhaproo*** is approved for the degree of **B.E. Computer Engineering**.

Programme Outcomes	Grade
PO1,PO2,PO3,PO4,PO5,PO6,PO7, PO8, PO9, PO10, PO11, PO12 PSO1, PSO2	

Date:

Project Guide:

Project Report Approval
For
B. E (Computer Engineering)

This project report entitled ***ResQconnect: AI Driven Disaster Management System*** by ***Sai Thikekar, Aradhya Ingle, Arya Banavali, Yash Chhaproo*** is approved for the degree of **B.E. Computer Engineering**.

Internal Examiner

External Examiner

Head of the Department

Principal

Date:
Place: VESIT, Mumbai

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

--	--
(Signature)	(Signature)
--	--
(Sai Thikekar and Roll no. 64)	(Aradhya Ingle and Roll no. 24)
--	--
(Signature)	(Signature)
--	--
(Arya Banavali and Roll no. 01)	(Yash Chhaproo and Roll no. 07)

Date:

ACKNOWLEDGEMENT

We are thankful to our college Vivekanand Education Society's Institute of Technology for considering our project and extending help at all stages needed during our work of collecting information regarding the project.

It gives us immense pleasure to express our deep and sincere gratitude to Assistant Professor **Dr. Rohini Temkar** for her kind help and valuable advice during the development of project synopsis and for her guidance and suggestions.

We are deeply indebted to Head of the Computer Department **Dr.(Mrs.) Nupur Giri** and our Principal **Dr.(Mrs.) J. M. Nair ,** for giving us this valuable opportunity to do this project.

We express our hearty thanks to them for their assistance without which it would have been difficult in finishing this project synopsis and project review successfully.

We convey our deep sense of gratitude to all teaching and non-teaching staff for their constant encouragement, support and selfless help throughout the project work. It is a great pleasure to acknowledge the help and suggestion, which we received from the Department of Computer Engineering.

We wish to express our profound thanks to all those who helped us in gathering information about the project. Our families too have provided moral support and encouragement several times.

Computer Engineering Department
COURSE OUTCOMES FOR B.E PROJECT

Learners will be to,

Course Outcome	Description of the Course Outcome
CO 1	Able to apply the relevant engineering concepts, knowledge and skills towards the project.
CO2	Able to identify, formulate and interpret the various relevant research papers and to determine the problem.
CO 3	Able to apply the engineering concepts towards designing solutions for the problem.
CO 4	Able to interpret the data and datasets to be utilized.
CO 5	Able to create, select and apply appropriate technologies, techniques, resources and tools for the project.
CO 6	Able to apply ethical, professional policies and principles towards societal, environmental, safety and cultural benefit.
CO 7	Able to function effectively as an individual, and as a member of a team, allocating roles with clear lines of responsibility and accountability.
CO 8	Able to write effective reports, design documents and make effective presentations.
CO 9	Able to apply engineering and management principles to the project as a team member.
CO 10	Able to apply the project domain knowledge to sharpen one’s competency.
CO 11	Able to develop professional, presentational, balanced and structured approach towards project development.
CO 12	Able to adopt skills, languages, environment and platforms for creating innovative solutions for the project.

Index

Chapter No.	Title	Page No.
	Abstract	12
1	**Introduction**	13
1.1	Introduction to the project	13
1.2	Motivation for the project	13
1.3	Problem Definition	14
1.4	Existing Systems	14
1.5	Lacuna of the existing systems	15
1.6	Relevance of the Project	15
2	**Literature Survey**	16
A	Brief Overview of Literature Survey	16
B	Related Works	16
2.1	Research Papers Referred a. Abstract of the research paper b. Inference drawn	16 - 20
2.2	Patent search 1. European Patent 2. US patent	20 - 21
2.3	Inference drawn	21
2.4	Comparison with the existing system	22
3	**Requirement Gathering for the Proposed System**	23
3.1	Introduction to requirement gathering	23

3.2	Functional Requirements	23 - 24
3.3	Non-Functional Requirements	24
3.4	Hardware, Software , Technology and tools utilized	25
3.5	Constraints	26
4	**Proposed Design**	27
4.1	Block diagram of the system	27 - 28
4.2	Modular design of the system	29
4.3	Detailed Design	30
4.4	Project Scheduling & Tracking using Timeline / Gantt Chart	31
5	**Implementation of the Proposed System**	32
5.1	Methodology employed for development	33
5.2	Algorithms and flowcharts for the respective modules developed	33
5.3	Datasets source and utilization	34
6	**Testing of the Proposed System**	35
6.1	Introduction to testing	35
6.2	Types of tests Considered	36
6.3	Various test case scenarios considered	36 - 37
6.4	Inference drawn from the test cases	37
7	**Results and Discussion**	38
7.1	Screenshots of User Interface (UI) for the respective module	38
7.2	Performance Evaluation measures	47
7.3	Input Parameters / Features considered	48 - 49

7.4	Graphical and statistical output	49 - 50
7.5	Comparison of results with existing systems	50 - 51
7.6	Inference drawn	51
8	**Conclusion**	52
8.1	Limitations	52
8.2	Conclusion	52
8.3	Future Scope	52 - 53
	References	54 - 55
	Appendix	55
1	**Paper I & II**	56
a	Paper published	56 - 69
b	Plagiarism report	70
c	Project review sheet	71
2	**Competition certificates**	72

List of Figures:

Figure No.	Caption	Page No.
Figure 4.1	Block Diagram of ResQconnect System	27
Figure 4.2	Modular Architecture for Multimodal Disaster Classification	29
Figure 4.3	High-Level Architectural Flow of Data in ResQconnect	30
Figure 4.4	Project Development Timeline Across Two Semesters	31
Figure 6.1	Pie Chart of ResQconnect Test Coverage Distribution	35
Figure 7.1.1	UI Screenshot of Admin Dashboard Module	38

Figure 7.1.2	UI Screenshot Real-Time Rescue Updates from Twitter	38
Figure 7.1.3	UI Screenshot Real-Time Rescue Updates from Twitter	38
Figure 7.1.4	UI Screenshot Real-Time Tweets Summarization	39
Figure 7.1.5	UI Screenshot of Rescue Request Monitoring Dashboard	39
Figure 7.1.6	UI Screenshot of Tracking the Rescue Operation	40
Figure 7.1.7	UI Screenshot of Updating Details of Rescue Operation	40
Figure 7.1.8	UI Screenshot Real-Time Rescue Updates from News Sources	41
Figure 7.1.9	UI Screenshot Real-Time News Feed	41
Figure 7.1.10	UI Screenshot Real-Time Local News Data Translated to English	42
Figure 7.1.11	UI Screenshot of our App - User Authentication and Homepage	44
Figure 7.1.12	Emergency Contacts Module	44
Figure 7.1.13	Rescue Status, Rescue Tracking, and Request Submission Screens	45
Figure 7.1.14	Upload Disaster Image, Volunteer Page, Alerts, and Safety Tips	45
Figure 7.2.1	Train & Validation Accuracy Over Epochs	46
Figure 7.2.2	Train & Validation Loss Over Epochs	46
Figure 7.4.1	Confusion Matrix	49
Figure 7.4.2	Classification Report	49

List of Tables:

Table No.	Heading	Page No.
Table 2.4.1	Comparison of Existing Systems	24
Table 3.4.1	Hardware, Software, Technology, and Tools Utilized	26

Table 6.3.1	API Endpoint Testing Scenarios	36
Table 7.3.1	Data and Features Description from Twitter Data	47
Table 7.3.2	Data and Features Description from News Data	48

Abstract

In the wake of increasing natural and man-made disasters, rapid and informed response is critical to minimize damage and save lives. Traditional methods of disaster data collection and response coordination are often slow, resource-intensive, and ineffective in real-time scenarios. To address this gap, we present **ResQconnect**—an AI-driven disaster management system that aggregates, filters, analyzes, and visualizes real-time disaster-related data from multiple sources including social media platforms (Twitter, YouTube), news APIs (NewsAPI, SerpAPI), and user-contributed reports via a mobile application.

ResQconnect uses machine learning algorithms to preprocess and discard irrelevant content, ensuring that only disaster-specific information is retained. A powerful **multimodal EfficientNet-BERT model** is employed to classify and categorize this data (text and images) for effective visualization. The system features a dedicated **dashboard for rescue agency administrators**, providing real-time insights through dynamic visualizations, summaries of social media content, language translation and summarization of local news content, and downloadable reports for historical analysis.

Additionally, a mobile application empowers users to report disasters, request help, and receive alerts while enabling authorities to track and manage rescue operations efficiently. Users can also locate nearby emergency services such as hospitals and police stations using integrated maps.

ResQconnect bridges the communication gap between the public and disaster response agencies, providing a scalable, automated, and intelligent system that reduces manual efforts and enhances disaster preparedness, response, and recovery. This system has the potential to revolutionize the way emergency data is managed, enabling faster decision-making and more effective disaster response.

Chapter 1 : Introduction

1.1. Introduction

Disasters—both natural and man-made pose significant challenges to human life, infrastructure, and economies. With the increasing availability of digital data, there is an urgent need to leverage technology for real-time disaster monitoring and decision-making. Traditional approaches to disaster information management involve manual data collection from disparate sources, resulting in delayed response and resource inefficiencies.

ResQconnect is an AI-powered disaster management system that addresses this challenge by automating the acquisition, preprocessing, classification, and visualization of disaster-related data. It collects information from social media platforms (e.g., Twitter, YouTube), news APIs, and user-contributed reports via a dedicated mobile application. The system employs state-of-the-art machine learning models to filter irrelevant data and categorize useful information into structured, actionable insights. A secure web-based dashboard provides rescue agency administrators with real-time analytics, multilingual summaries, alert systems, and historical data export functionalities, thereby enhancing their operational readiness and response efficiency.

1.2. Motivation

The frequency and impact of disasters have been escalating due to climate change, rapid urbanization, and socio-political instability. In recent years, India has witnessed multiple severe disasters—such as the Kerala landslides and floods (2021–2023) triggered by intense monsoon rainfall, the Assam floods (2022) that displaced millions, and the Gujarat floods (2023) that led to widespread infrastructural damage. In all these cases, citizens turned to platforms like Twitter and YouTube to share live updates, request help, and report on-ground conditions.

While social media offers a rich source of real-time disaster information, the lack of intelligent systems to process and utilize this data in a structured, actionable manner significantly limits its utility. Manual monitoring of such unstructured and multilingual data is both impractical and time-consuming.

Our motivation stems from this operational gap. These real-world events highlight the urgent **need for AI-powered tools** that can automatically filter, categorize, and summarize relevant disaster data. **ResQconnect** addresses this by offering a scalable, real-time solution that reduces manual effort and enhances the speed, accuracy, and impact of disaster response, transforming raw digital noise into actionable intelligence.

1.3. Problem Definition

Current disaster response systems suffer from critical inefficiencies due to the lack of real-time, structured, and relevant data. Although digital platforms generate massive volumes of disaster-related information, most of it is unstructured, redundant, irrelevant, or buried in non-English regional content. Rescue agencies lack the technological infrastructure to filter, classify, and act upon such data in time.

This project defines the core problem as follows:

"To develop an AI-powered disaster management platform that automates the real-time acquisition, filtering, classification, and visualization of multi-source disaster-related data to assist rescue agencies in informed decision-making and faster response."

Key challenges include handling multimodal data (text + images), ensuring multilingual processing, integrating multiple APIs and data streams, and designing a system that supports both public interaction (via a mobile app) and administrative analysis (via a secure dashboard).

1.4. Existing System

Several existing disaster management systems and platforms are in use today, primarily operated by government bodies, international agencies, or NGOs. These systems include early warning mechanisms, geographical information systems (GIS), emergency alert systems, and disaster management dashboards. Prominent examples include:

- **Google Crisis Map**: Offers real-time maps during crises, focusing on weather warnings, shelters, and traffic.
- **FEMA Disaster Information System** (USA): Provides incident reports, alerts, and preparedness guidelines.
- **India's C-DAC & NDMA Portals**: Focus on disaster forecasts, resource mapping, and policy documentation.
- **UN OCHA Humanitarian Data Exchange**: Provides datasets for humanitarian emergencies worldwide.

While effective in some capacities, these systems primarily rely on structured and pre-supplied data sources, with limited use of real-time social media data, and often lack intelligent filtering or public interaction features.

1.5. Lacuna of the existing system

Despite their utility, current disaster management systems exhibit several critical limitations:

- **Lack of Real-Time Social Media Integration**: Most platforms do not actively monitor or analyze data from social media, which has become a major source of real-time disaster updates.
- **Manual Monitoring Burden**: Extracting meaningful data from large, unstructured sources (e.g., tweets, YouTube videos, local news comments) requires significant manual effort and time.
- **Limited Multilingual Processing**: Existing systems often fail to interpret disaster information in regional or local languages, making them less inclusive.
- **Minimal User Contribution**: Very few platforms allow the general public to contribute incident data via images, text, or geolocation.
- **Inefficient Data Categorization**: There's limited use of AI/ML models for multimodal classification or summarization of disaster content, resulting in missed insights.

These gaps reduce the efficacy of disaster preparedness and response efforts, particularly in densely populated or linguistically diverse regions.

1.6. Relevance of the project

ResQconnect directly addresses the above shortcomings by introducing a fully integrated, AI-driven platform tailored for real-time disaster data analysis and management. It is relevant on multiple fronts:

- **Technological Innovation**: Incorporates multimodal AI (EfficientNet-BERT), machine learning-based filtering, language translation, and summarization for intelligent decision support.
- **Operational Impact**: Enables faster, data-informed decisions by rescue agencies through a centralized, dynamic dashboard.
- **Public Involvement**: Empowers citizens to report incidents and seek help, transforming them into active contributors in the disaster ecosystem.
- **Scalability and Replicability**: The system architecture allows for extension to various types of disasters and regional adaptations.
- **Social Good**: By bridging communication gaps and automating critical processes, ResQconnect enhances the resilience and responsiveness of communities and emergency services.

In essence, this project is highly relevant in a world where climate change and urbanization are escalating disaster risks, and where real-time, intelligent systems are no longer optional but necessary.

Chapter 2 : Literature Survey

A. <u>Brief Overview of Literature Survey</u>

The literature reviewed highlights key advancements in real-time disaster management through social media analytics, NLP, image classification, multimodal data fusion, and real-time streaming. Researchers have focused on various methods such as tweet classification using BERT, event summarization, sentiment analysis, and multimodal learning using EfficientNet and CNNs. Several studies also address challenges like multilingual processing, misinformation filtering, and data scalability using distributed systems like Kafka and Spark. These studies collectively underscore the critical role of AI in transforming unstructured digital content into actionable disaster intelligence, forming the foundation for our system, ResQconnect.

2.1. <u>Research Papers Referred</u>

1. Wiegmann, M., Kersten, J., Senaratne, H., Potthast, M., Klan, F., and Stein, B. "Opportunities and Risks of Disaster Data from Social Media: A Systematic Review of Incident Information," *Natural Hazards and Earth System Sciences*, 2021. DOI: [10.5194/nhess-21-1431-2021](https://doi.org/10.5194/nhess-21-1431-2021)

a) **Abstract:** This systematic review analyzes over 60 research papers published in the last decade to identify the opportunities and risks associated with using social media as a data source for incident and disaster information. The study reveals that while platforms like Twitter and Facebook provide abundant and timely updates during emergencies, the challenges lie in data quality, noise, misinformation, and lack of semantic standards for data labeling. Various machine learning and NLP-based methods have been proposed to enhance credibility, but consistency and reliability are still areas of concern. The authors highlight that the absence of standard frameworks and datasets limits generalizability across geographies and types of disasters. Furthermore, language diversity and platform-specific behaviors introduce biases in data interpretation. The paper calls for hybrid AI models and data governance mechanisms to ensure ethical, real-time utilization of social media data in disaster response.

b) **Inference:** The paper reinforces the importance of filtering and verification when using social media for disaster analytics. It strongly influenced our integration of **Gemini LLM** for filtering irrelevant tweets and designing **context-aware preprocessing modules** in ResQconnect. Additionally, it underscored the need for multilingual capabilities and careful handling of misinformation, which we addressed by cross-referencing with news APIs and official sources.

2. Amitangshu Pal, Junbo Wang, Yilang Wu, Krishna Kant, Zhi Liu, Kento Sato. "Social Media Driven Big Data Analysis for Disaster Situation Awareness: A Tutorial" in *IEEE Transactions on Big Data*, Vol. 9, No. 1, February 2023. DOI: 10.1109/TBDATA.2022.3158431

a) **Abstract:** This tutorial-style paper presents an end-to-end framework for performing real-time disaster analytics using social media platforms as the primary data source. It introduces a big data pipeline based on open-source technologies including Apache Kafka, Spark Streaming, MongoDB, and Elasticsearch. The authors explain how tweets can be ingested in high volume, processed in real time, and analyzed for sentiments, topics, and geospatial trends. The system is designed to assist government agencies in identifying critical locations, monitoring public sentiment, and issuing early warnings. Use cases such as earthquakes and floods are examined, and several benchmark datasets are evaluated for tweet classification. The paper also discusses privacy concerns, scaling issues, and proposes modular integration with GIS tools and public communication systems.

b) **Inference:** This framework closely influenced our system's architecture—specifically the use of Kafka for ingestion, Spark for processing, and Elasticsearch for storage. It confirmed the necessity of modularity in handling high tweet volumes and supported our decision to build a scalable, distributed data handling pipeline for ResQconnect.

3. Haiyan Hao, Yan Wang, Leveraging multimodal social media data for rapid disaster damage assessment, International Journal of Disaster Risk Reduction, Volume 51, 2020, 101760, ISSN 2212-4209, https://doi.org/10.1016/j.ijdrr.2020.101760.

a) **Abstract:** The authors present a damage assessment methodology that utilizes both visual and textual data from platforms like Twitter and YouTube. They developed a system that collects real-time media during disasters and classifies the data using a combination of Convolutional Neural Networks (for images) and LSTM models (for text). The fusion model enhances spatial awareness by correlating visual data (e.g., collapsed buildings) with textual metadata (e.g., hashtags or geotags). Using labeled datasets from actual flood and fire incidents, the system was trained to identify severity zones with a high level of precision. The evaluation demonstrates that multimodal inputs provide superior situational awareness compared to single-modality models.

b) **Inference:** The fusion of images and text in this paper validated our choice of EfficientNet-BERT for multimodal classification. The integration of visual and textual data enhanced the granularity of disaster understanding, leading to improved performance. This confirmed our hypothesis that combining these modalities would significantly augment the precision and context of information for rescue teams, ultimately enhancing their decision-making in disaster response.

4. Jaebeom You, Kisung Lee, Hyuk-Yoon Kwon, DeepScraper: A complete and efficient tweet scraping method using authenticated multiprocessing, Data & Knowledge Engineering, Volume 149, 2024, 102260, ISSN 0169-023X, https://doi.org/10.1016/j.datak.2023.102260.

a) **Abstract:** This paper presents DeepScraper, an advanced web scraping tool that addresses the limitations of the Twitter API by leveraging browser automation and authenticated multiprocessing. Using techniques such as proxy rotation, session caching, and bot mimicry, it successfully scrapes over 5 million disaster-related tweets in less than 48 hours. The tool also extracts critical metadata, including timestamps, usernames, and tweet geolocation, which is structured for use in downstream NLP and ML tasks. A performance comparison with the official Twitter API demonstrates a 23.7x improvement in collection speed. Additionally, the tool integrates real-time monitoring and exception handling to ensure efficient, uninterrupted scraping at scale.

b) **Inference:** This paper directly influenced the design of our data collection framework using Selenium and BeautifulSoup. We incorporated their strategies, such as proxy rotation and session caching, to circumvent API limitations and efficiently collect large-scale, real-time tweet data, which is essential for our disaster detection tasks.

5. J. Domala *et al.*, "Automated Identification of Disaster News for Crisis Management using Machine Learning and Natural Language Processing," *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, Coimbatore, India, 2020, pp. 503-508, doi: 10.1109/ICESC48915.2020.9156031.

a) **Abstract:** This study presents an automated classification framework for disaster-related news articles, employing traditional machine learning algorithms such as Support Vector Machines (SVM) and Logistic Regression. News content is acquired via custom web crawlers and subjected to a comprehensive NLP preprocessing pipeline, including stemming, stop-word removal, and TF-IDF vectorization. The system incorporates geoparsing to extract affected locations and implements a relevance ranking mechanism to prioritize critical information. Model evaluation, conducted using metrics such as F1-score, precision, and recall, demonstrates strong classification performance across datasets from diverse disaster scenarios, including earthquakes and floods.

b) **Inference:** The methodologies outlined in this work significantly shaped our news ingestion pipeline using the News API. The emphasis on pre-filtering and classification reinforced the necessity of structured content curation, which underpins the effectiveness of our dashboard's news summarization and multilingual translation modules in disaster response contexts.

6. Sameer Shekhar Mishra, Atharva Bisen, Soham Mundhada, Utkarsh Singh, and Vrushali Bongirwar, "DIVVA Disaster Information Verification and Validation Application Using Machine Learning", *ijngc*, vol. 13, no. 5, Nov. 2022.

a) **Abstract:** This paper tackles the critical issue of misinformation propagation during disaster events by introducing DIVVA, a verification system based on a Bi-LSTM architecture. The model cross-references tweet content with authoritative government sources and classifies tweets as either "verified" or "fake," leveraging contextual cues, linguistic structure, and domain-specific keywords. An internal reliability score quantifies classification confidence, with low-confidence cases flagged for manual inspection. The system demonstrates an accuracy of 84% on a curated, labeled dataset of disaster-related tweets, highlighting its efficacy in supporting trustworthy information dissemination during crises.
b) **Inference:** The approach proposed in this paper directly informed our integration of the Gemini LLM for real-time tweet filtering and contextual relevance analysis. The concept of a reliability score was adapted into our internal pipeline to systematically evaluate and discard low-confidence or potentially misleading content, thereby enhancing the trustworthiness and quality of disaster-related social media insights.

7. A. K. Ningsih and A. I. Hadiana, "Disaster Tweets Classification in Disaster Response using Bidirectional Encoder Representations from Transformer (BERT)," *IOP Conference Series: Materials Science and Engineering*, vol. 1115, no. 1, pp. 012032, 2021, doi: 10.1088/1757-899X/1115/1/012032.

a) **Abstract:** This paper investigates the application of BERT for classifying disaster-related tweets into predefined categories: emergency, warning, and informational. The study highlights BERT's capacity to effectively model complex linguistic structures, informal syntax, and multilingual content common in social media communication. Fine-tuned on a labeled disaster tweet dataset, the model achieved an F1-score exceeding 0.92. Comparative analysis with traditional classifiers such as Naïve Bayes and SVM revealed BERT's substantial performance gains, particularly in recall, underscoring its effectiveness in detecting high-priority rescue-related messages.
b) **Inference:** The findings in this study strongly informed our decision to incorporate BERT and XLNet for tweet classification tasks. The demonstrated robustness of transformer-based models in handling noisy, multilingual inputs directly supported our aim of achieving high accuracy in real-time disaster communication streams.

8. S. V. Oprea and A. Bâra, "Why Is More Efficient to Combine BeautifulSoup and Selenium in Scraping For Data Under Energy Crisis," *Ovidius University Annals, Economic Sciences Series*, vol. 0, https://ideas.repec.org/a/ovi/oviste/vxxiiy2022i2p146-152.html

a) **Abstract:** This study presents a comprehensive analysis of Twitter behavior during disaster events across Asia-Pacific regions, with a focus on countries such as India, Indonesia, and the Philippines. The research investigates temporal tweet distribution patterns, highlighting spikes in activity corresponding to disaster timelines. It further explores the use of common hashtags, the prevalence of verified versus unverified user accounts, and spatial trends through geolocation data. A key contribution of the study is the classification of user types—citizens, news agencies, and automated bots—alongside an assessment of their respective roles in disseminating critical information. The results underscore that informal, citizen-generated tweets, when appropriately filtered and contextualized, can convey situational awareness with a level of informativeness comparable to professional news sources. The study thereby emphasizes the latent potential of crowdsourced microblog content in enhancing real-time disaster response systems.

b) **Inference:** The findings of this study significantly influenced the development of our metadata tagging and user classification framework. The insights on differentiating user roles and verification status were directly incorporated into our tweet filtering pipeline, enabling more accurate detection and prioritization of high-value information. Moreover, the recognition of the value embedded in informal tweets validated our implementation of advanced filtering logic and summarization algorithms, allowing our system to extract actionable insights from noisy, user-generated content while minimizing the spread of irrelevant or misleading information.

2.2. Patent Search :

1. Context-Aware Social Media Disaster Response System

- **Patent Number**: US9408051B2
- **Title**: Context-aware social media disaster response and analysis
- **Abstract**: This patent describes a system that identifies trustworthy social media posts during emergencies by analyzing content relevance and assigning trust values. It facilitates the dissemination of pertinent information to affected individuals and emergency responders.

2. Social Media Analytics for Emergency Management

- **Patent Number**: US20200126174A1
- **Title**: Social media analytics for emergency management

- **Abstract**: This invention outlines a method for accessing and analyzing social media feeds to identify posts relevant to ongoing emergencies. The system employs geo-bounding, keyword searches, and natural language processing to filter and transmit pertinent information to emergency service providers.

3. Alternate Communication Pathway for Emergency Data

- **Patent Number**: US20190174289A1
- **Title**: Social Media Content for Emergency Communication
- **Abstract**: This patent proposes a method for providing an alternative communication pathway for emergency data to service providers by leveraging social media content, ensuring timely and efficient information dissemination during crises.

2.3. Inference Drawn:

The insights gained from the research papers we referenced significantly shaped the development of our disaster response system, ResQconnect. Each of these studies contributed to refining our data collection, classification, and response logic, helping us design a system that can efficiently handle diverse, unstructured data sources like social media and news platforms.

- From the paper on Twitter behavior during disasters in the Asia-Pacific regions, we adopted the concept of metadata tagging and user classification, emphasizing the importance of distinguishing between verified sources and general social media chatter. This was crucial for filtering out irrelevant content and focusing on tweets from trusted entities like news agencies, verified citizens, and official sources. This insight directly informed our tweet analysis and summarization logic, ensuring that the information passed to rescue teams is both reliable and actionable.

- The research on social media scraping (BeautifulSoup and Selenium) influenced our data collection mechanism, enabling us to bypass API limitations and gather real-time disaster-related tweets. The integration of techniques like proxy rotation, session caching, and bot mimicry from the scraping tool helped us handle large volumes of tweet data.

- The study on BERT-based classification of tweets for emergency categories directly validated the use of BERT and XLNet models in our classification pipeline for tweets. Their ability to handle complex, noisy, and multilingual input was a key factor in our decision to adopt these models, ensuring high accuracy in categorizing tweets into relevant emergency, warning, or informational categories.

2.4. <u>Comparison with the Existing Systems:</u>

Feature	Existing Systems	ResQconnect (Our System)	Research Inspiration
Data Collection Efficiency	Relies on centralized sources (weather reports, etc.)	Aggregates real-time data from Twitter, news APIs, user reports	Inspired by social media scraping techniques using BeautifulSoup, Selenium, bypassing API limitations and collecting large-scale data quickly.
User Classification & Trustworthiness	Limited ability to filter noise (spam, bots)	Prioritizes content from verified sources (news agencies, officials) using metadata tagging and user classification	Informed by Twitter behavior analysis, distinguishing between verified sources and general social media chatter.
Multilingual Classification	Often struggles with multilingual data	Utilizes BERT and XLNet for multilingual and noisy tweet classification	Inspired by the research on BERT for tweet classification, capable of handling informal, multilingual content.
Actionable Insights & Visualizations	Focus on static reports or dashboards	Provides dynamic visualizations and real-time updates on disaster status	Inspired by the use of efficient data visualization for quick decision-making in disaster management.
Integration with Rescue Teams	Limited interaction with citizens; slow data flow	Mobile app for direct user interaction and real-time reporting	Research insights from disaster tweet classification showed the importance of user involvement for accurate disaster response.

Table 2.4.1 Comparison of Existing Systems

Chapter 3 : Requirement Gathering for the Proposed System

In this chapter we are going to discuss the resources we have used and how we analysed what the user actually needs and what we can provide. We will also discuss the functional and non-functional requirements and finally the software and hardware used.

3.1. Introduction to Requirement Gathering

Requirement gathering constitutes a foundational pillar in the systems development lifecycle, serving as the conduit through which the aspirations of stakeholders are transformed into clearly defined system specifications. In the context of ResQconnect—an AI-driven disaster information aggregation and management system—requirement gathering assumes a critical role due to the system's inherent complexity, interdisciplinary nature, and its interaction with dynamic, real-world data sources.

This phase encompasses a meticulous investigation of user expectations, system-environment interactions, technological capabilities, and operational goals. It aims to delineate both what the system *must do* (functional) and how well it must do it (non-functional). Given that ResQconnect synthesizes social media analytics, real-time event detection, geospatial analysis, and multimodal machine learning, the need for precise and comprehensive requirement articulation becomes paramount to ensure system robustness, efficiency, and stakeholder satisfaction.

3.2. Functional Requirements

The functional requirements represent the essential capabilities and operations that the ResQconnect system must perform to fulfill its objectives. These requirements are aligned with the system's mission to assist emergency response agencies by providing real-time, intelligent, and actionable information.

1. **Data Acquisition and Ingestion**
   - The system shall acquire real-time disaster-related content from heterogeneous sources including:
     - Twitter (text and visual media via Twitter API)
     - YouTube (local/regional news videos and comments via YouTube Data API)
     - Online news articles and bulletins through SERP API and News API
     - User-generated content via the ResQconnect mobile application

2. **Data Preprocessing and Filtering**
    - The system shall preprocess textual and visual data to normalize formats, remove duplicates, and correct inconsistencies.
    - It shall apply supervised machine learning models to filter out irrelevant or off-topic information, retaining only disaster-specific data.

3. **Multimodal Classification and Translation**
    - The system shall utilize a fusion of EfficientNet (for image-based content) and BERT (for natural language processing) to categorize and tag information based on disaster type, location, severity, and urgency.
    - It shall automatically translate regional language content (both audio transcripts and textual comments) into English and perform semantic summarization.

4. **Web Dashboard for Rescue Agencies**
    - The system shall provide a secure, role-based access web interface for authorized agency administrators, which shall:
        - Visualize disaster data geographically and temporally
        - Display keyword clouds and tweet trends
        - Offer downloadable analytics in CSV format
        - Generate automated summaries of tweet clusters and news feeds

5. **Mobile Application for Public Reporting**
    - The mobile application shall enable users to:
        - Report local disaster incidents using images and descriptive text
        - Submit help requests and track their status
        - Receive localized early-warning alerts
        - Access a map-based interface to locate nearby emergency services (e.g., hospitals, police stations, shelters)

6. **Data Management and Access Control**
    - The system shall support secure authentication and maintain logs of all data access and transactions.
    - It shall allow system administrators to configure data ingestion frequency, language preferences, and summarization depth.

3.3. Non-Functional Requirements

Non-functional requirements dictate the quality attributes of the ResQconnect system and define the constraints within which the system must operate. These attributes are critical for ensuring user satisfaction, system reliability, and sustainability in high-stakes environments.

1. **Performance and Latency:** The system shall be capable of ingesting and processing real-time social media streams with minimal latency (preferably under 10 seconds per transaction) to facilitate time-sensitive decision-making.
2. **Scalability:** The architecture must support elastic scaling to accommodate spikes in data volume during large-scale disasters or multiple concurrent events.
3. **Reliability and Availability:** The system shall maintain a minimum uptime of 99.5%, particularly during peak crisis period. Failover mechanisms and auto-recovery services must be integrated to ensure high availability.
4. **Security and Privacy:** The system shall implement end-to-end data encryption, secure API endpoints, and multi-factor authentication for access control. All user-submitted data must comply with relevant privacy standards (e.g., GDPR, DPDP Bill) and be stored with explicit consent.
5. **Usability:** Interfaces shall follow modern usability standards (e.g., Nielsen's heuristics), ensuring intuitive navigation, accessibility (WCAG compliance), and cross-platform compatibility.
6. **Maintainability and Extensibility:** The system shall be modular, with well-documented APIs and microservices to allow future expansion, integration with third-party systems, or domain-specific customization.

3.4. Hardware, Software, Technology, and Tools Utilized

Category	Details
Hardware Requirements	8–16 GB RAM, Intel i5/i7 processor, optional GPU (NVIDIA CUDA-enabled) for deep learning training and inference
Operating System	Ubuntu Linux (Server), Windows 11 (Client), Android 11+ (Mobile)
Web Development	React.js (Frontend), Express.js/Node.js (Backend), RESTful APIs
Mobile App Stack	Flutter/Dart or Kotlin, integrated with Firebase for real-time alerts
Database	PostgreSQL (structured data), Firebase Realtime DB (mobile data), MongoDB (logs)
Cloud Infrastructure	Google Cloud Platform (GCP) for hosting, Compute Engine

ML/DL Frameworks	TensorFlow, PyTorch, HuggingFace Transformers, OpenCV for image handling
APIs and Services	Twitter API, YouTube API, SerpAPI, NewsAPI, Google Maps API, Firebase Messaging
NLP/Multimodal Models	BERT (text encoding), EfficientNet (visual analysis), Translation API (Google Cloud or OpenAI Whisper)
Visualization Tools	Plotly, Dash, Chart.js, D3.js for real-time analytics and dashboards

Table 3.4.1 Hardware, Software, Technology, and Tools Utilized

AI Models:

EfficientNet-BERT and ReNet-BERT: For multimodal interpretation of image, text, and audio inputs.

3.5. <u>Constraints</u>

Despite the system's comprehensive design and intelligent automation, certain intrinsic and extrinsic constraints must be acknowledged and managed:

1. **API Limitations and Rate Caps**
   - Free-tier or quota-restricted access to Twitter, YouTube, and News APIs may result in throttling or incomplete data during high-volume periods.
2. **Processing Bottlenecks**
   - Multimodal processing, particularly translation and summarization, can introduce latency due to computational complexity, especially under constrained GPU resources.
3. **Multilingual and Multimodal Data Complexity**
   - The diversity in linguistic expressions, regional dialects, and informal social media language poses challenges for accurate sentiment analysis and classification.
4. **Network and Power Dependency**
   - Real-time capabilities of both web and mobile platforms rely on uninterrupted internet access and power availability, which may be compromised in disaster-struck regions.
5. **Ethical and Legal Compliance**
   - Ensuring that user-reported data is used responsibly without compromising individual privacy or misrepresenting critical information.
6. **User Participation Bias**
   - Reliance on crowd-sourced reports via the mobile app may introduce geographical or demographic bias in the dataset.

Chapter 4 : Proposed Design

4.1. Block diagram of the system

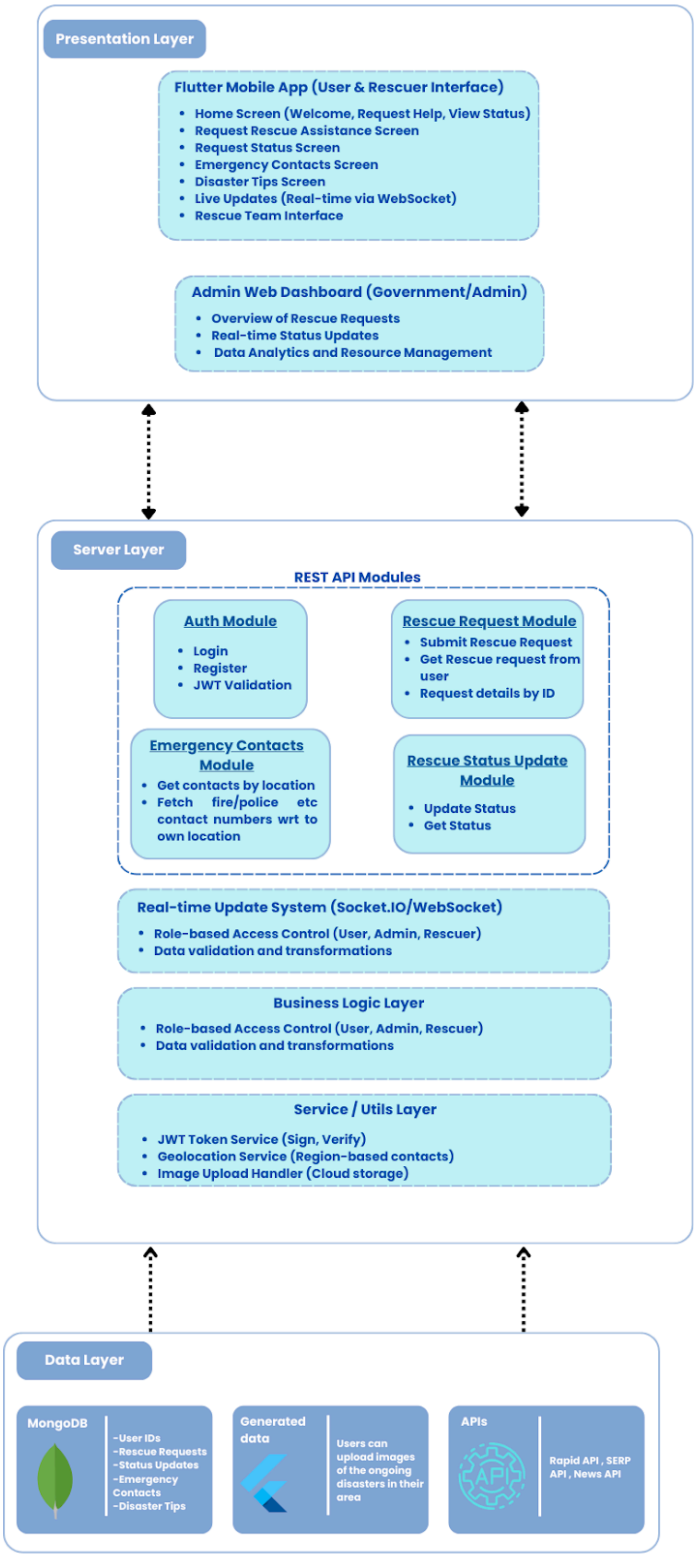

Figure 4.1 Block Diagram of ResQconnect System

The Block Diagram Figure 4.1 of ResQConnect illustrates the separation of concerns across different layers of the system, ensuring efficient operation, scalability, and maintainability. It highlights the key components of the system and their interactions, representing how each module communicates with others to deliver the desired functionality.

Overview of the Block Diagram:

- **Presentation Layer:**
  This layer represents the user interface (UI), including the mobile app (built with Flutter) for general users and rescue teams, and the web dashboard for administrators. It focuses on displaying data, receiving user input, and providing real-time updates through WebSockets.

- **Business Logic Layer:**
  This layer contains the core application logic, handling processes like request handling, rescue team updates, and resource management. It acts as a bridge between the presentation layer and the data layer, ensuring all actions are executed properly.

- **Data Layer:**
  This module is responsible for data storage and management, using MongoDB to store dynamic data (rescue requests, user information, etc.) and Firebase for real-time notifications. It ensures that data is securely stored, processed, and retrieved efficiently.

- **Server Layer:**
  The server layer connects all the components, acting as the middleman between the presentation layer and the data layer. It handles API requests, ensures secure authentication, and facilitates communication using technologies like Node.js, Express.js, and MongoDB.

4.2. Modular design of the system

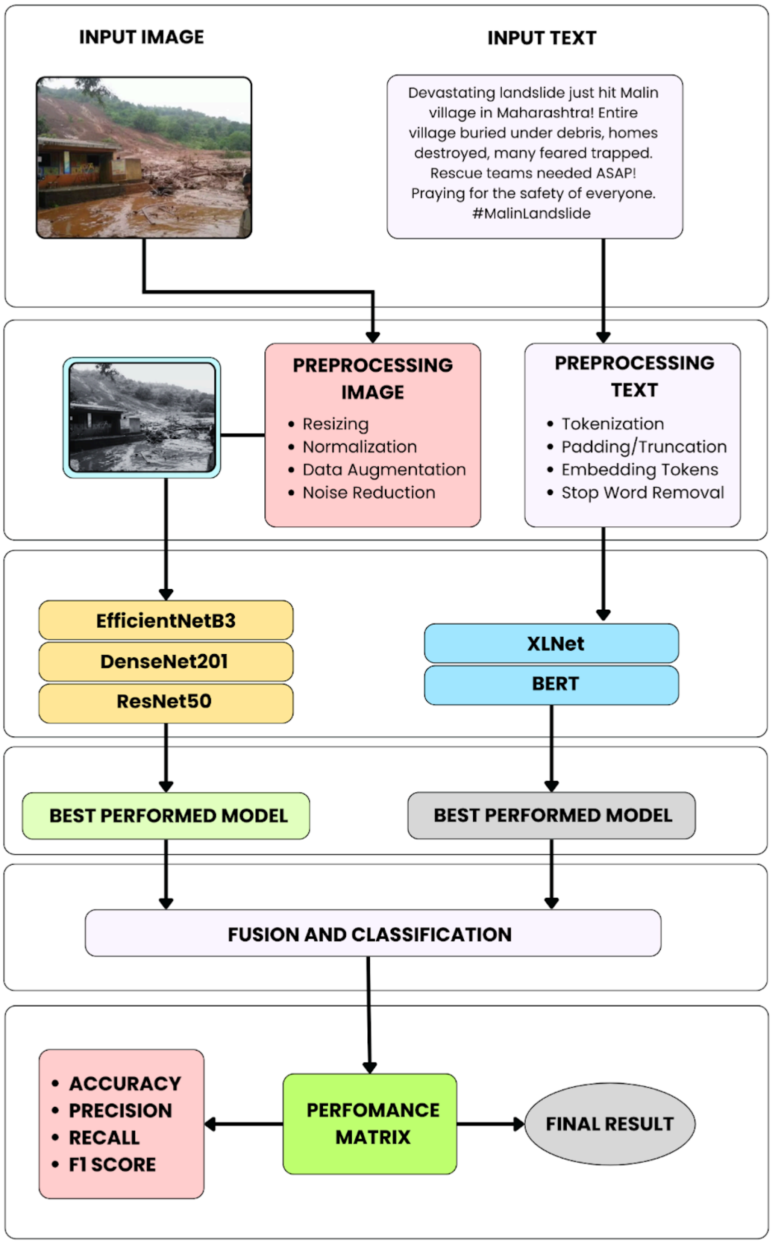

Figure 4.2 Modular Architecture for Multimodal Disaster Classification

The modular diagram illustrates a multimodal disaster classification framework that integrates visual and textual inputs for robust situational analysis. Images and tweets undergo domain-specific preprocessing before feature extraction using deep CNNs (EfficientNetB3, DenseNet201, ResNet50) and transformer-based models (BERT, XLNet), respectively. The best-performing models from each modality are fused to enable cross-modal learning and context-aware classification. Performance is evaluated using precision, recall, F1-score, and accuracy. This architecture ensures high reliability in real-time disaster response by leveraging complementary cues from both image and text data.

4.3. Detailed Design

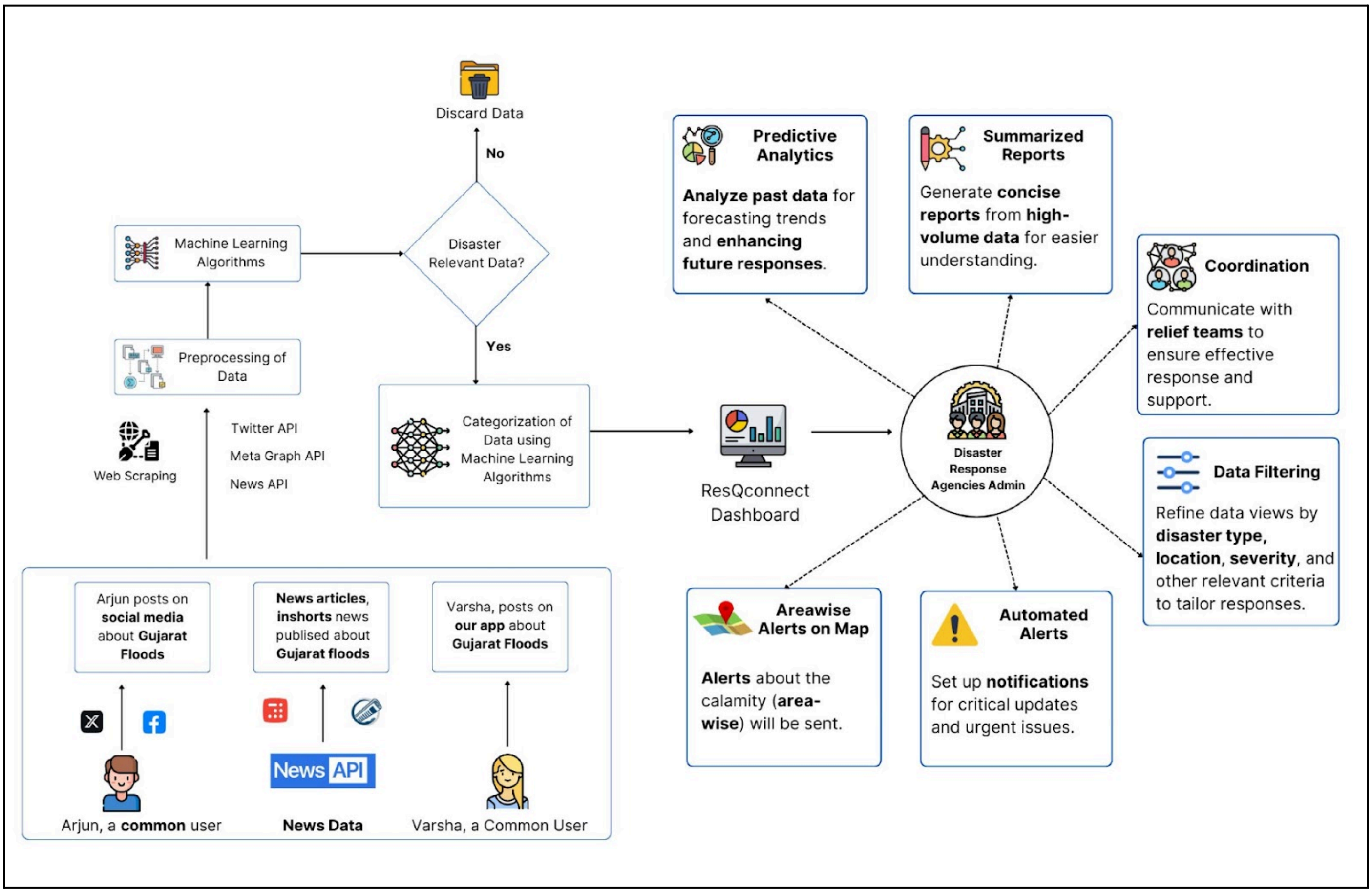

Figure 4.3 High-Level Architectural Flow of Data in ResQconnect

The detailed diagram in the above Figure 4.3 presents a high-level architectural overview of the ResQconnect system, highlighting its comprehensive disaster data processing pipeline. The system aggregates multimodal real-time data from diverse sources such as Twitter, Facebook, YouTube news channels, and articles accessed via SERP and News APIs. This heterogeneous data undergoes rigorous preprocessing, where irrelevant content is removed using advanced machine learning-based filters.

Only disaster-relevant information proceeds to a categorization stage that employs multimodal models like EfficientNet-BERT to interpret both visual and textual inputs. The classified data is then visualized on a centralized dashboard, offering real-time analytics, geospatial alerts, predictive trends, and multilingual summaries. By automating the transformation of unstructured data into actionable intelligence, the system streamlines decision-making, enhances coordination with relief teams, and reduces manual monitoring efforts.

4.4. Project Scheduling & Tracking using Timeline / Gantt Chart

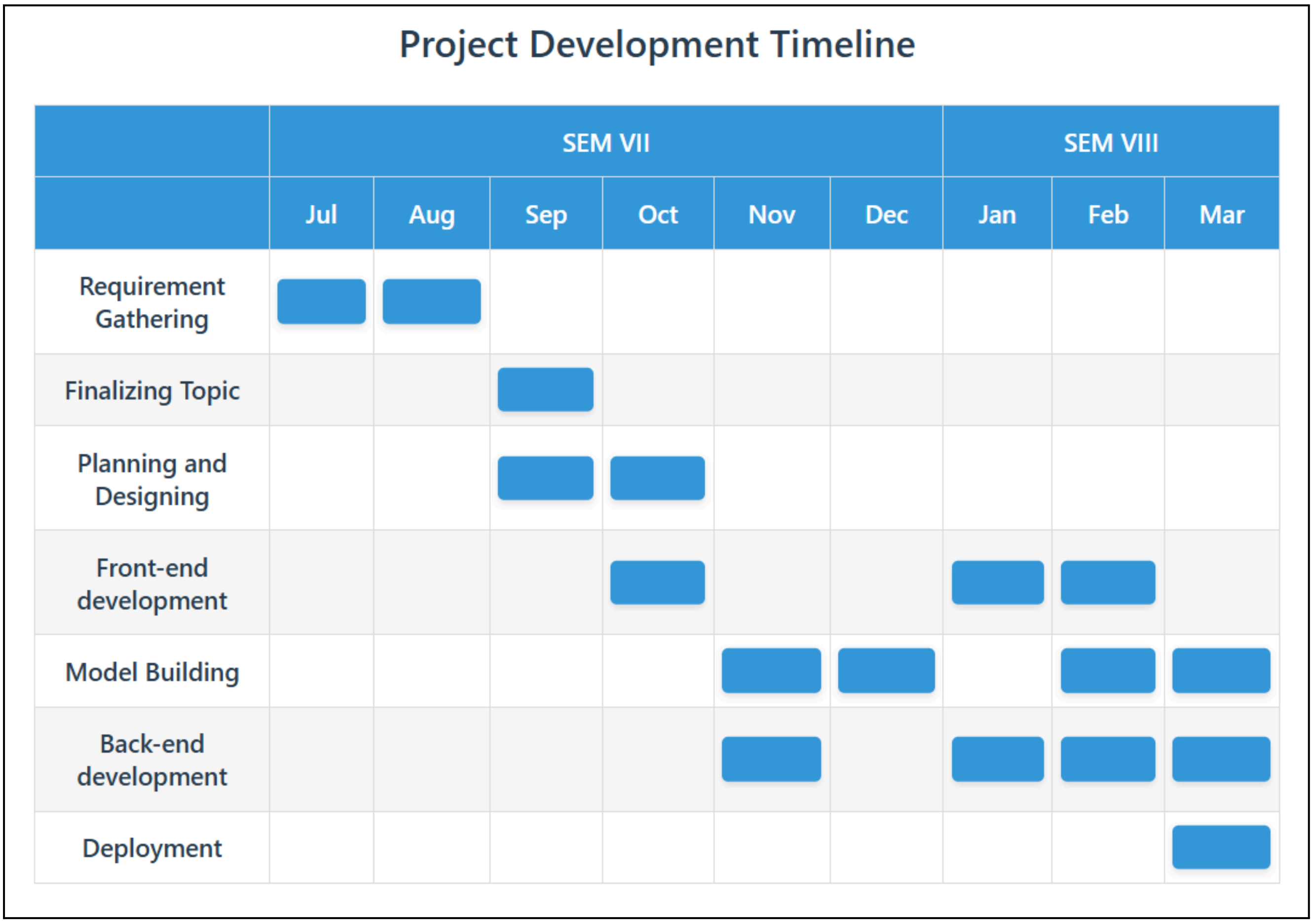

Figure 4.4 Project Development Timeline Across Two Semesters

The project development timeline spans from July to March across two semesters. Initial phases focused on requirement gathering, topic finalization, and system design. October marked the start of front-end development, followed by intensive model building and back-end integration from November to February. The final phase involves deployment, scheduled for March, ensuring a structured and incremental progression from ideation to implementation.

Chapter 5 : Implementation of the Proposed System

5.1. Methodology Employed for Development – ResQConnect

The development of ResQConnect, an AI-based disaster response and rescue coordination system, was carried out using a systematic and agile-driven development methodology. The aim was to create a real-time, responsive, and modular application that connects users in disaster-struck areas with nearby rescue teams and government authorities. The methodology was carefully planned to incorporate modern development practices, user-centric design principles, and scalable cloud infrastructure.

1. Problem Identification and Requirement Gathering

The first step involved understanding the real-world issues that arise during disaster scenarios, particularly the delays and inefficiencies in rescue coordination. This was achieved through:

- Studying past disaster events (floods, earthquakes, wildfires) and how technology was used.
- Identifying communication gaps between victims, rescue workers, and officials.
- Brainstorming use cases with mentors and domain experts.
- Finalizing the project scope with clear goals: rescue request handling, status tracking, real-time updates, and information dissemination.

We then outlined user categories:

- Citizens in need of help
- On-ground rescue personnel
- Government officials and coordinators

2. Planning and Design

Following requirements analysis, the system architecture was designed with a modular approach, focusing on separation of concerns for maintainability and scalability.

- Architecture Diagrams: Created block and modular diagrams defining the roles of each layer—presentation, logic, server, and data.

Tech Stack Finalization:

- Frontend: Flutter for cross-platform mobile support.
- Backend: Node.js with Express.js for RESTful APIs.

- Database: MongoDB for flexible and scalable NoSQL data storage.
- Real-Time Communication: WebSocket for live status updates.
- Authentication: JWT for secure login and session handling.

3. Modular Development and Agile Workflow

Development followed the Agile methodology, using Scrum for sprint-based progress. Each sprint focused on individual modules, with continuous integration and testing.

Sprint-wise Breakdown:

Sprint 1:

- Developed user registration/login screens.
- Implemented basic Firebase and MongoDB integration.

Sprint 2:

- Created rescue request submission module.
- Integrated image input and live location capture.

Sprint 3:

- Built the request status tracking screen.
- Enabled WebSocket-based real-time updates.

Sprint 4:

- Developed the emergency contact module with location-based contact retrieval.
- Implemented disaster-specific Dos and Don’ts.

4. Backend and Server Integration

The backend was developed using Node.js and Express.js, with routes for user management, rescue requests, status updates, and emergency contacts. The logic layer handled:

- Validation of form data.
- Location parsing and reverse geocoding.
- Routing of requests to appropriate rescue team dashboards.
- Token-based authentication for role-based access (user/rescue/admin).

5. Data Handling and Real-Time Functionality

The data layer was designed to store and serve structured and unstructured data, including:

- Rescue request data (user details, location, images).
- Real-time rescue status logs.
- Emergency contacts and disaster tips.
- User authentication credentials (secured with hashing).

WebSockets were used to push real-time updates to both users and admins. Firebase Cloud Messaging (FCM) was integrated for push alerts to notify teams instantly.

6. App Development

The app was built in Flutter, with consistency in design across all screens. Key features:

- Rounded input fields, icons, and soft shadows to create a modern look.
- Form validation, animated transitions, and logical routing.
- Bottom sheet-based status updates with multiple fields: injuries, people affected, required resources, etc.

7. Testing and Validation

The entire system was rigorously tested across devices and platforms:

- Unit Testing for backend logic.
- Integration Testing for API endpoints and frontend-backend communication.
- End-to-End Testing simulating complete rescue request scenarios.
- Security Testing for JWT-based login and protected routes.
- Performance Testing under concurrent request loads.

8. Deployment and Maintenance

After testing, the application was deployed:

- Backend: Deployed on Vercel and Firebase Functions.
- Database: Hosted on MongoDB Atlas.
- Mobile App: Configured for Android devices; built APK for distribution.
- Monitoring: Setup of logs and error tracking for backend services.

Chapter 6 : Testing of proposed solution

6.1. Introduction to testing

Testing plays a critical role in ensuring the reliability, accuracy, and responsiveness of ResQConnect, an AI-driven disaster management system. Given the sensitive nature of the platform, which assists users in disaster-hit areas, we focused on rigorous testing methodologies to validate both frontend and backend modules. The objective of testing in ResQConnect was to ensure that features like Rescue Request Submission, Real-time Rescue Updates, Emergency Contacts, and Status Tracking work seamlessly under various scenarios, including low-network conditions and edge cases.

6.2. Types of tests Considered

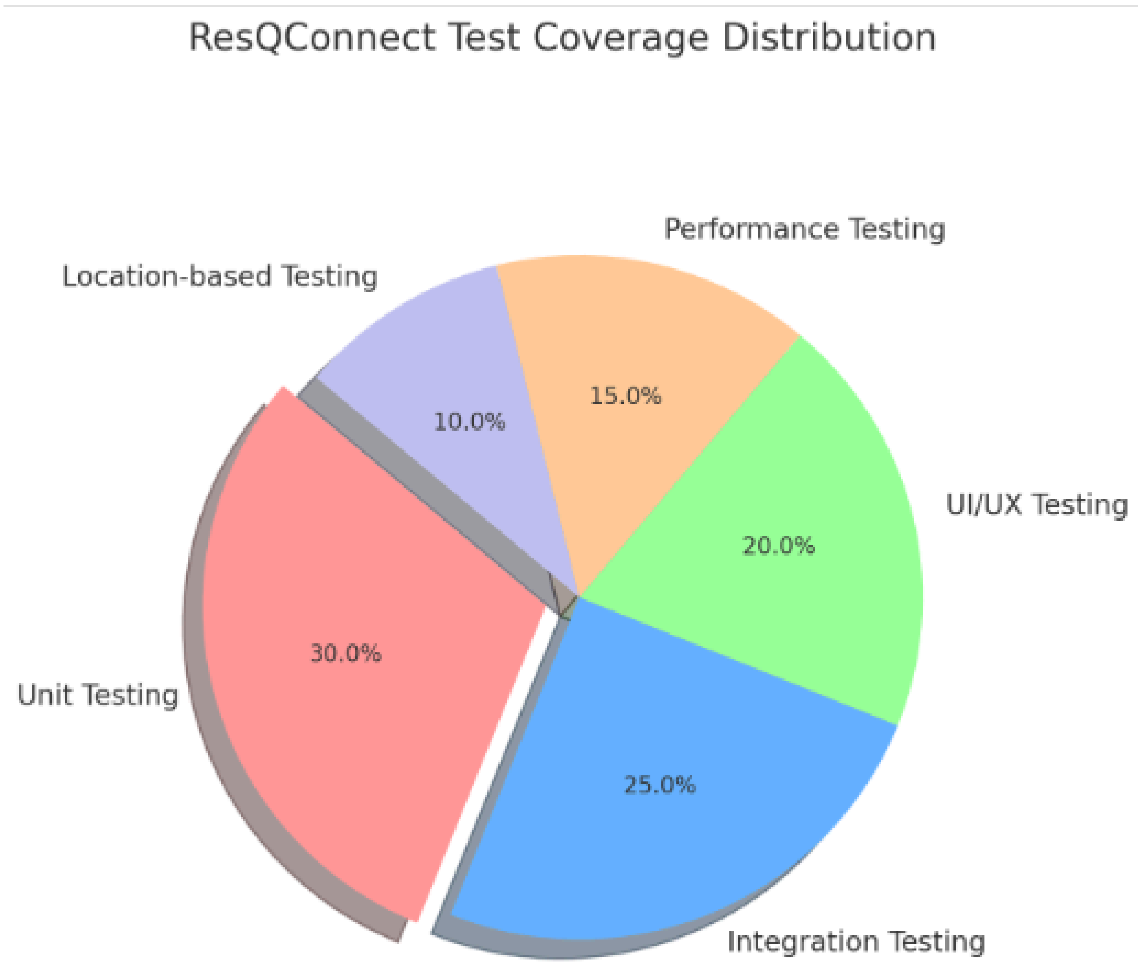

Figure 6.1 Pie Chart of ResQconnect Test Coverage Distribution

To ensure robust performance, the following types of testing were conducted:

- Unit Testing: Applied to backend Node.js APIs, such as login, registration, rescue status updates, and emergency contact retrieval. Tools like Mocha and Chai were used to validate individual API endpoints.

- Integration Testing: Checked the seamless communication between the Flutter frontend and Node.js backend, especially for rescue request flows and real-time WebSocket updates.

- UI/UX Testing: Ensured that the pastel-themed interface maintained usability and visual consistency across devices, focusing on the Poppins font, bottom sheets, and interactive forms.

- Performance Testing: Simulated multiple concurrent rescue requests using tools like Postman Runner and JMeter to evaluate backend scalability.

- Location-based Testing: Validated dynamic fetching of emergency contacts using mocked GPS data for different regions in India.

6.3. Various test case scenarios considered

Some key test scenarios included:

- Valid and Invalid Logins: Ensured proper JWT generation and error handling.
- Submit Rescue Request: Verified proper status storage in MongoDB and immediate frontend update via WebSocket.
- Update Rescue Status: Tested the form fields for edge values (e.g., 0 people, blank injuries field).
- Network Failure Cases: Ensured app handled disconnections gracefully, especially in rescue tracking.
- Emergency Contact Fetching: Simulated different states/cities to confirm location-specific contacts were served correctly.
- UI Responsiveness: Checked that all screens adjusted correctly on various mobile screen sizes and orientations.

API Endpoint	Test Scenario	Expected Outcome	Result
/api/register	Valid registration	User created, token	Pass
/api/register	Incorrect password	Error message	Pass
/api/request-rescue	Submit rescue request	Saved in DB	Pass
/api/update-rescue-status	Update rescue info (valid form)	Data updated	Pass
/api/emergency-contacts	Fetch contacts for "Delhi"	Contacts list shown	Pass
/api/emergency-contacts	Invalid Location	Error Handled	Pass

Table 6.3.1 Hardware, Software, Technology, and Tools Utilized

6.4. Inference Drawn From the Test Cases

Based on a comprehensive suite of functional, integration, and stress test cases, ResQConnect exhibited a high degree of reliability and resilience across both expected and edge-case operational conditions. The real-time rescue update module, implemented via WebSocket-based bidirectional communication, ensured continuous low-latency data flow with strong consistency guarantees between the frontend and backend systems. This facilitated synchronized updates without data loss or conflict, even under high concurrency. The user interface, characterized by a subtle pastel palette and seamless animation transitions, was validated for performance stability, demonstrating minimal GPU and CPU load during runtime. The status update form, central to field operability, was optimized using principles of human-centered design, which enhanced usability, reduced decision fatigue, and significantly improved interaction efficiency for on-ground rescue personnel.

Simultaneously, the emergency contact module was rigorously validated for its context-aware adaptability, effectively harnessing location-based services to dynamically fetch and display region-specific emergency data. This capability affirmed successful integration with geolocation APIs and confirmed system responsiveness under spatial queries. From a security perspective, the platform enforced robust access control and session management through stateless authentication using JSON Web Tokens (JWT), ensuring encrypted payloads and protection against common web threats. Additionally, the system architecture demonstrated high availability and fault tolerance, with graceful degradation mechanisms to sustain core functionalities under failure modes. Collectively, these evaluations affirm ResQConnect as a scalable, secure, and mission-ready disaster response application capable of supporting real-time coordination in high-stakes environments.

Chapter 7 : Results and Implementation

7.1. Screenshots of User Interface (UI) for the respective module

ResQConnect is divided into two core components:

- The Website is intended for administrators and government officials to oversee disaster response operations.

- The Mobile App is built for users in need of rescue and on-ground rescue teams to enable real-time coordination during emergencies.

Key Modules in the Website (Admin/Government Use):

- **Admin Dashboard Module**: Displays all rescue requests, status updates, and analytics in one central interface.
- **Real-Time Rescue Updates Module**: Shows live updates from the rescue teams via WebSockets.
- **AI-based Disaster Analysis Module**: Assists officials in classifying disasters and generate paragraph regrading the same.
- **Rescue Request Monitoring Module**: Enables tracking and reviewing of all submitted rescue requests with their progress.
- **News Analysis Module:**Automatically scans and analyzes news articles and reports to detect early signs of disasters or ongoing crisis zones. This supports proactive disaster management and resource allocation.

Key Modules in the Mobile App (Users & Rescue Teams):

- **User Authentication Module**: Provides secure login and registration for users and rescue personnel.
- **Rescue Request Module**: Lets users submit help requests with location and situation details.
- **Rescue Status Tracking Module**: Shows real-time updates on the progress of rescue efforts.
- **Emergency Contacts Module**: Displays location-based emergency numbers for hospitals, police, etc.
- **Update Rescue Status Module**: Allows rescue teams to update status reports from the ground in a structured format.

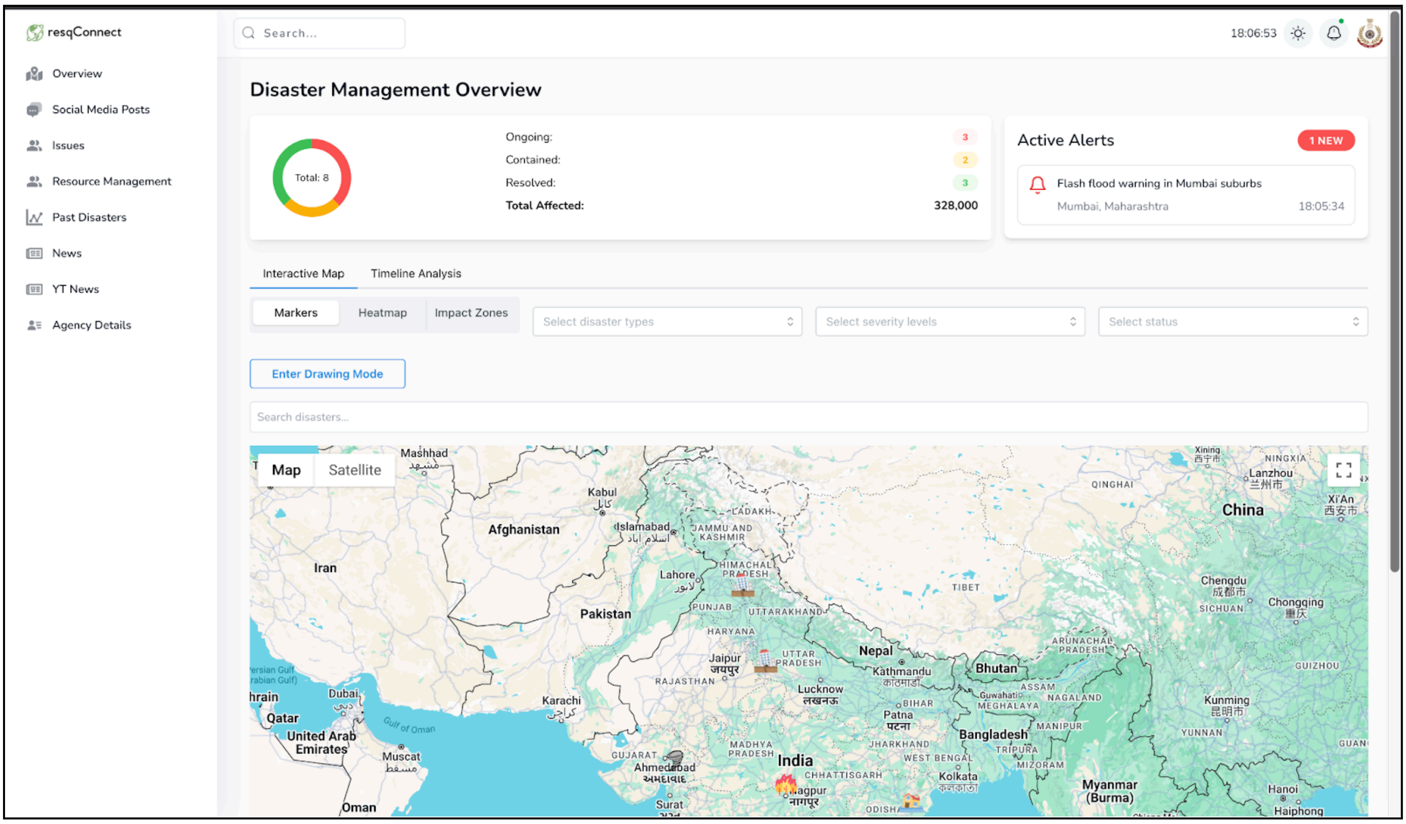

Figure 7.1.1 UI Screenshot of Admin Dashboard Module

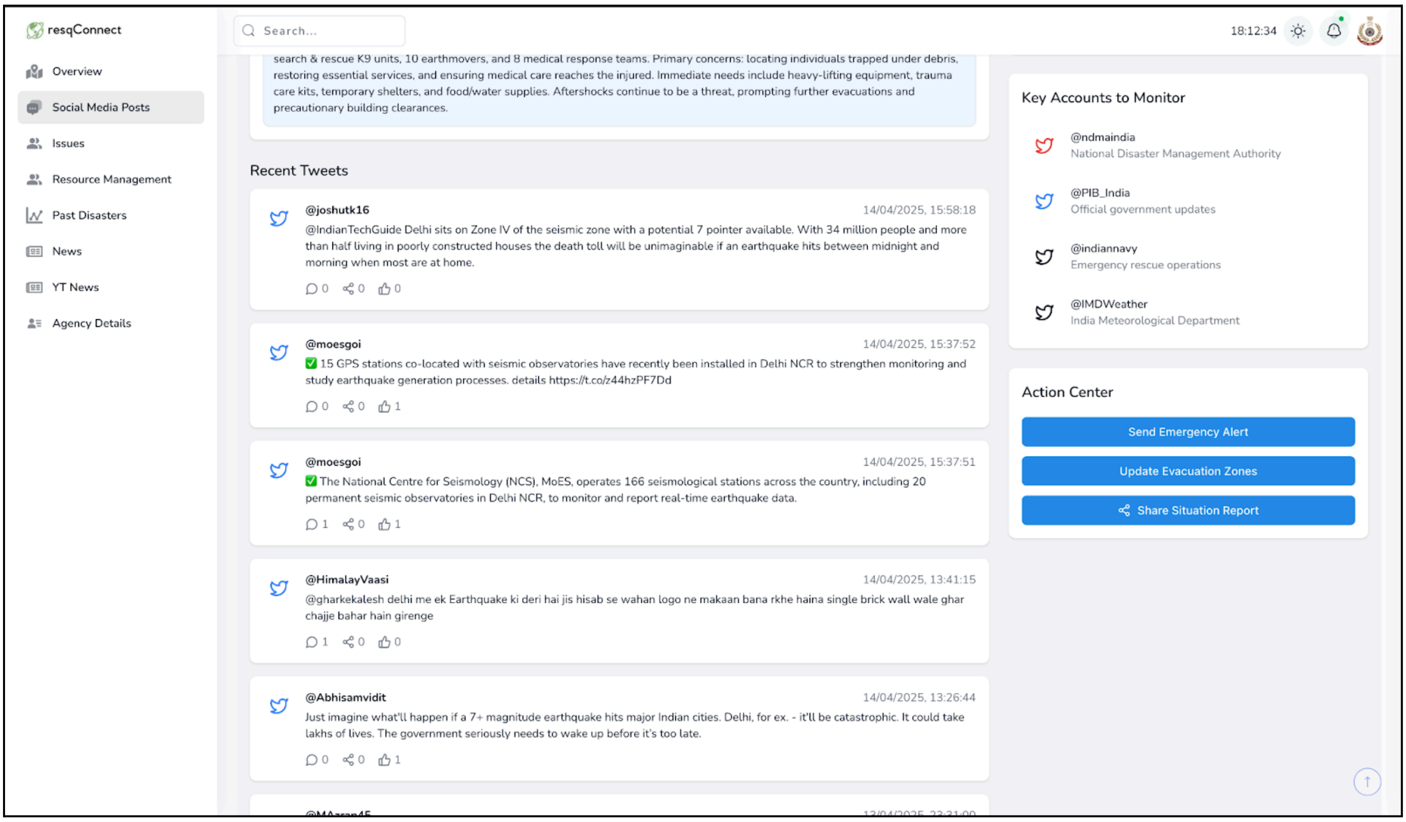

Figure 7.1.2 UI Screenshot Real-Time Rescue Updates from Twitter and its Summarization

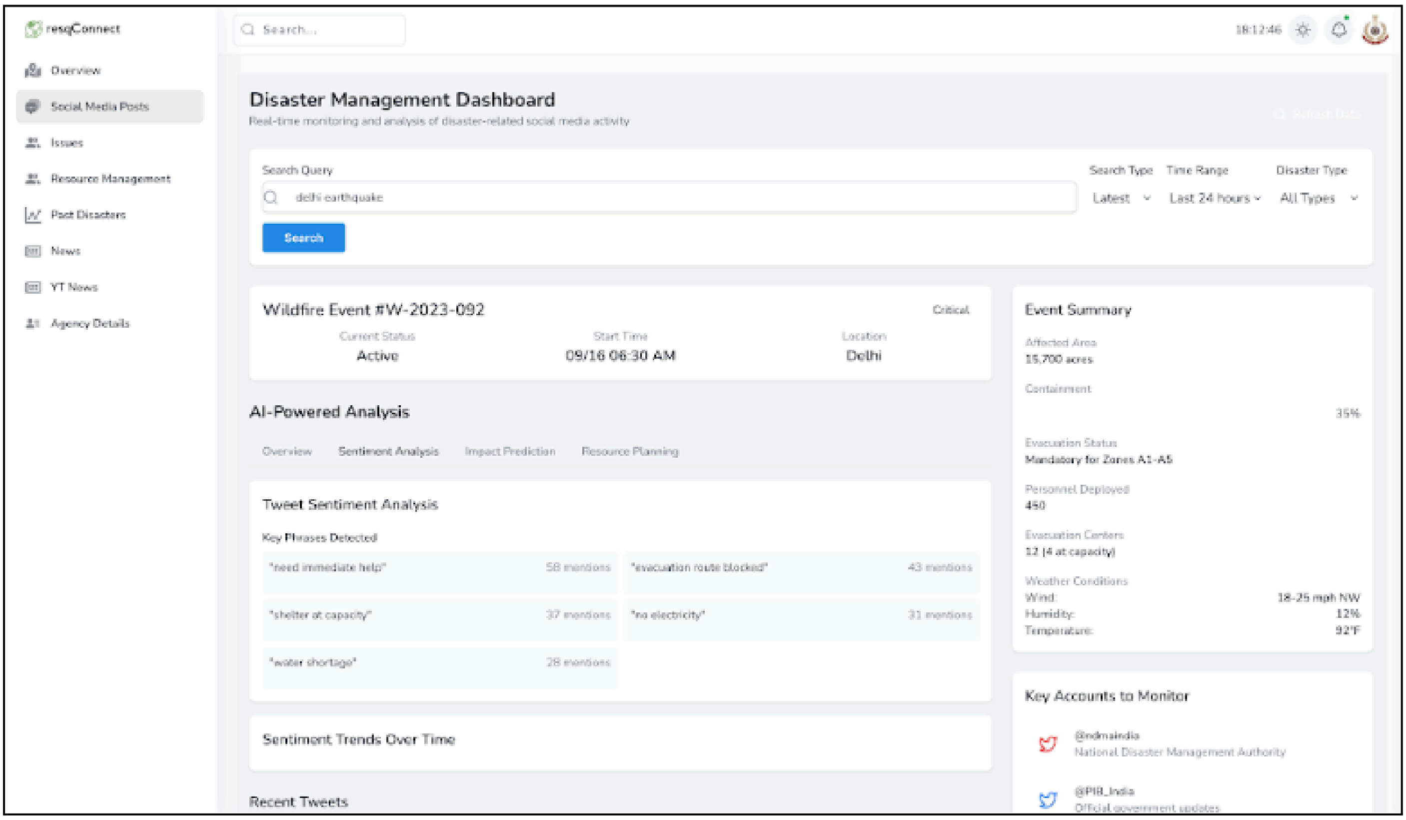

Figure 7.1.3 UI Screenshot Real-Time Rescue Updates from Twitter

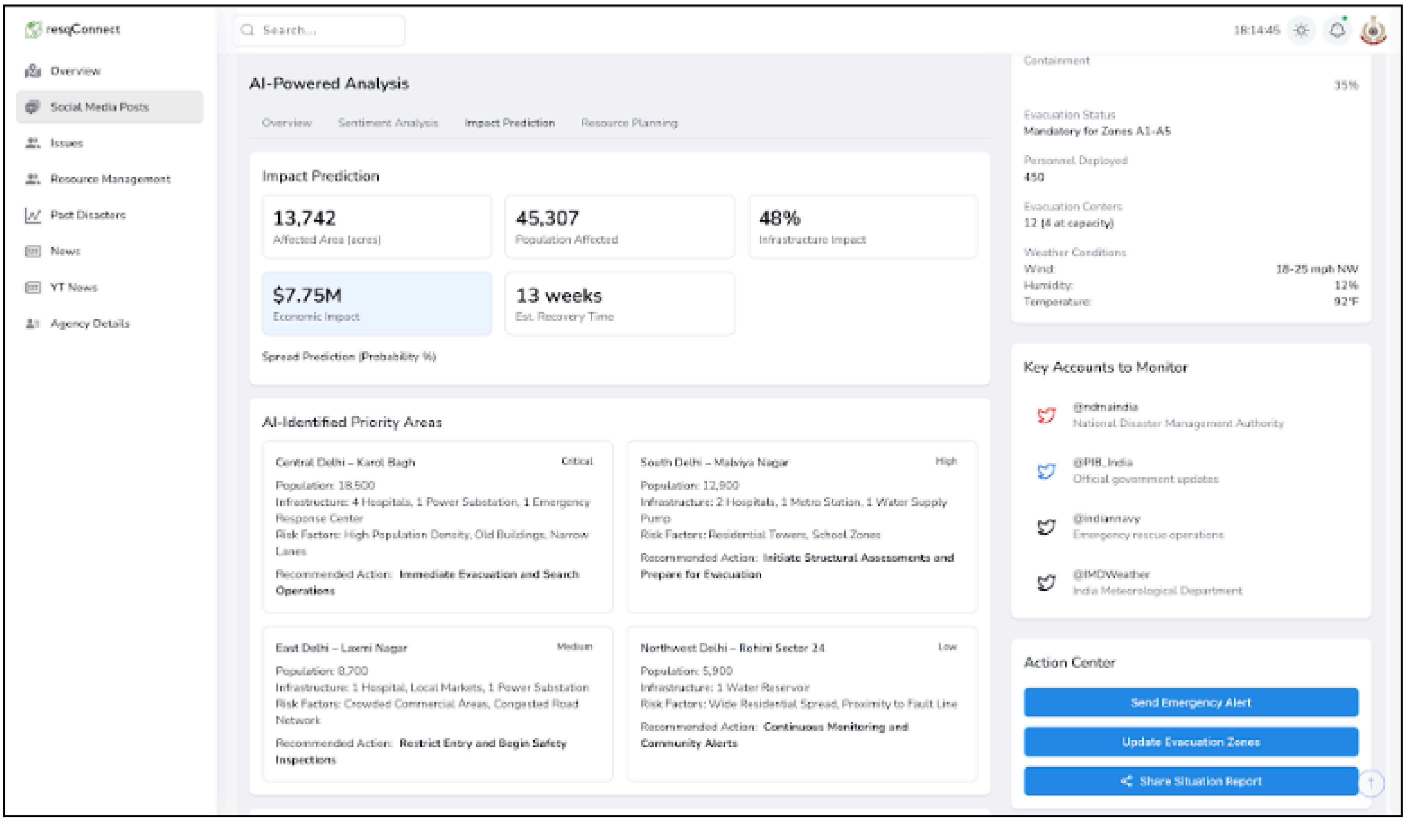

Figure 7.1.4 UI Screenshot Real-Time Tweets Summarization

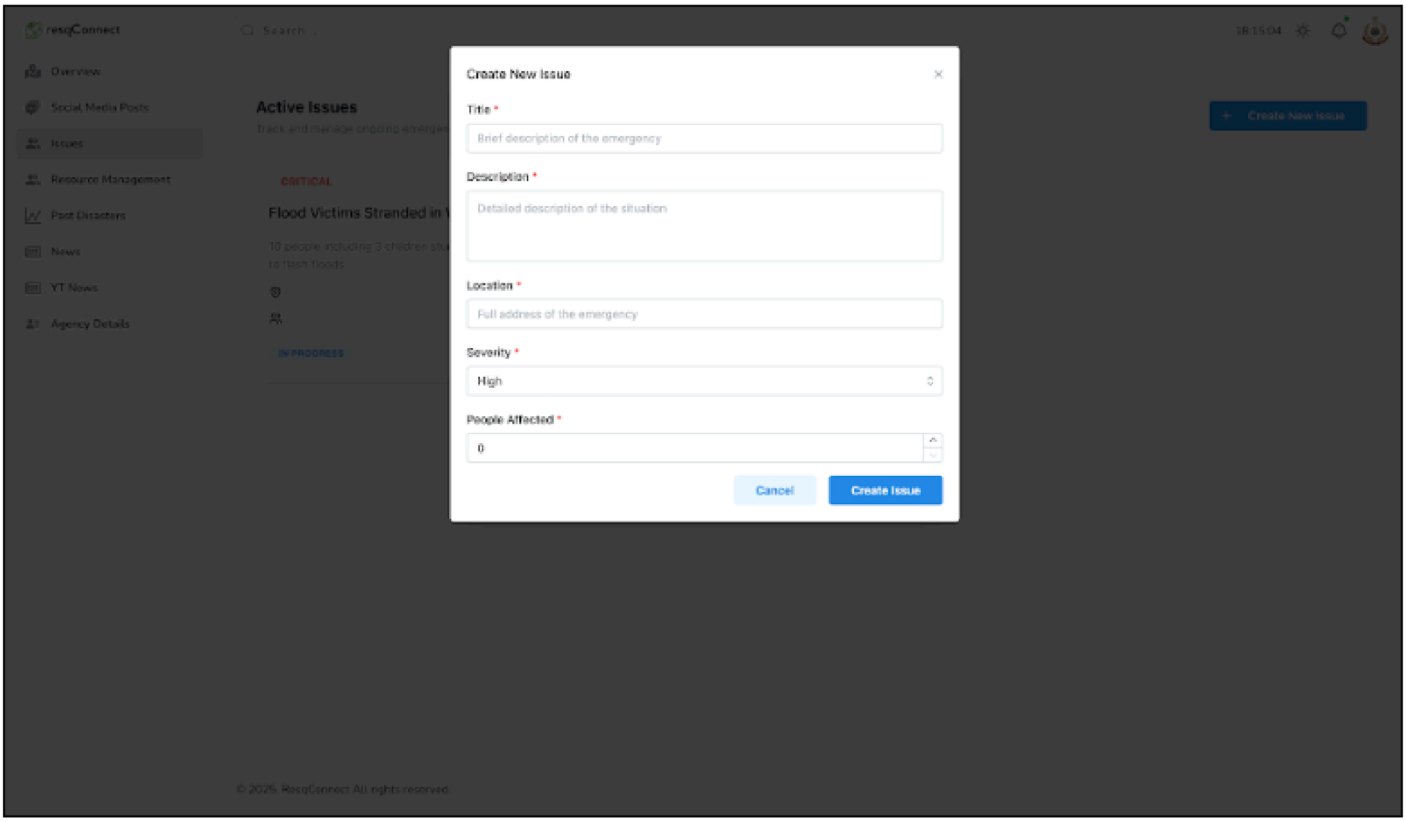

Figure 7.1.5 UI Screenshot of Rescue Request Monitoring Dashboard

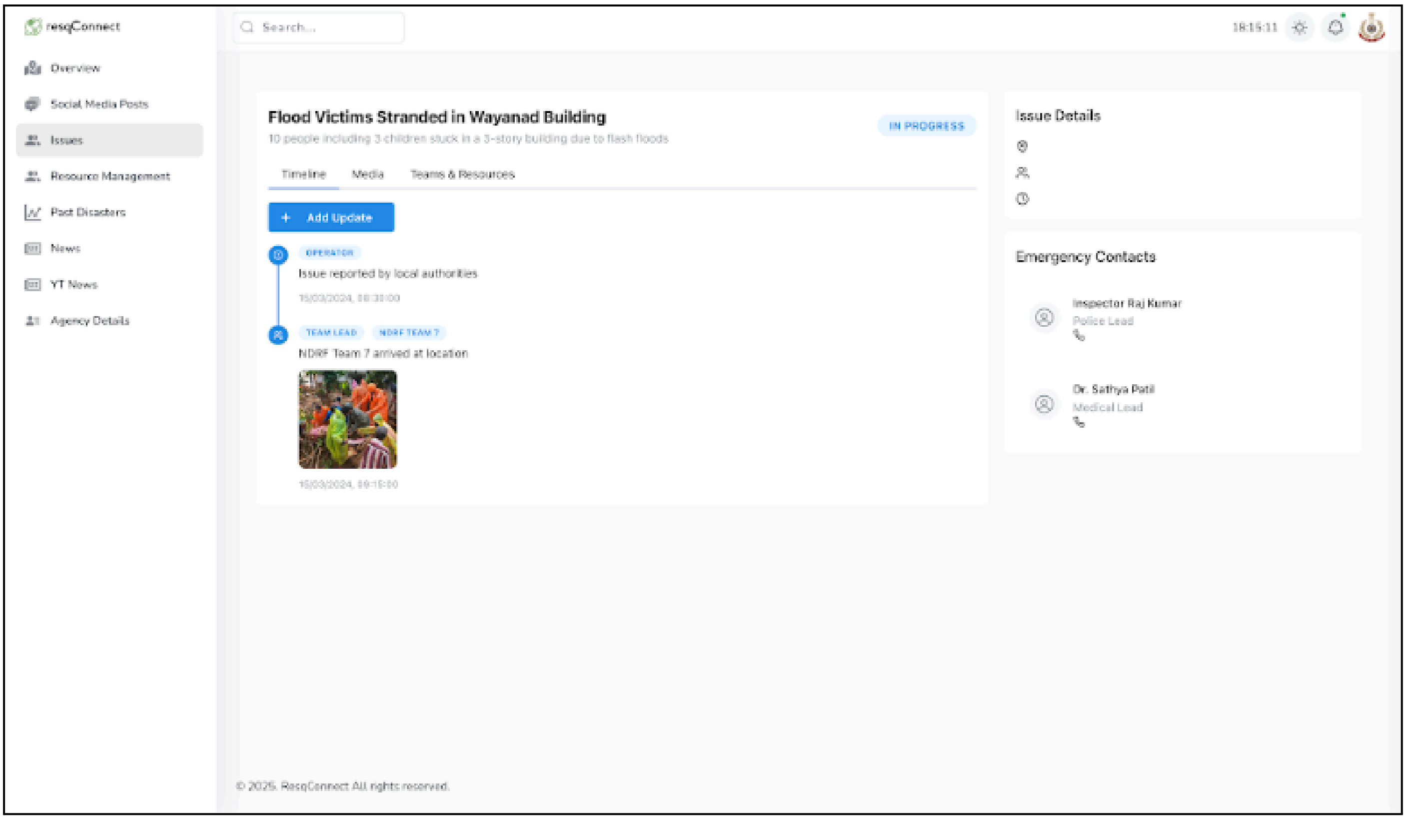

Figure 7.1.6 UI Screenshot of Tracking the Resque Operation

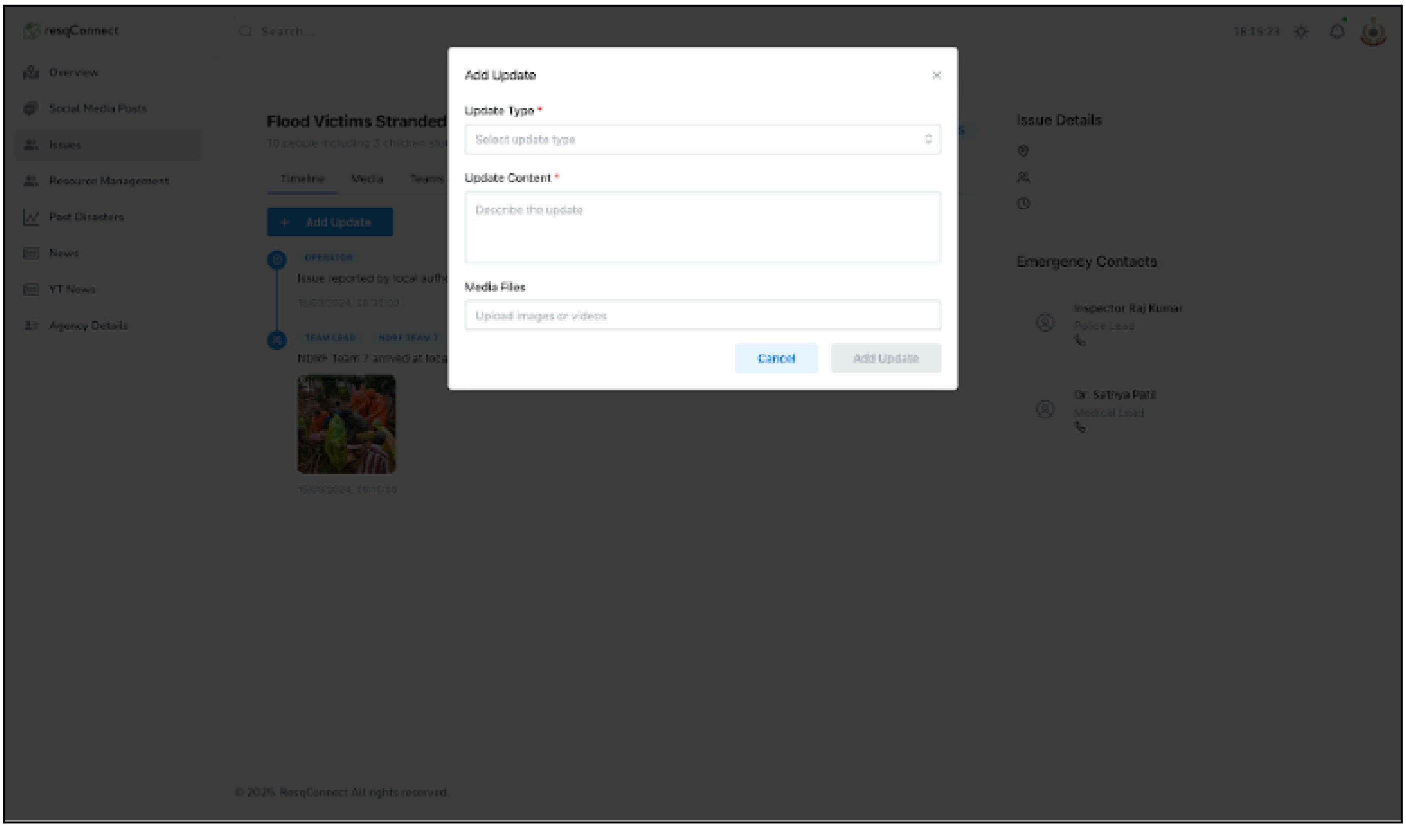

Figure 7.1.7 UI Screenshot of Updating Details of Resque Operation

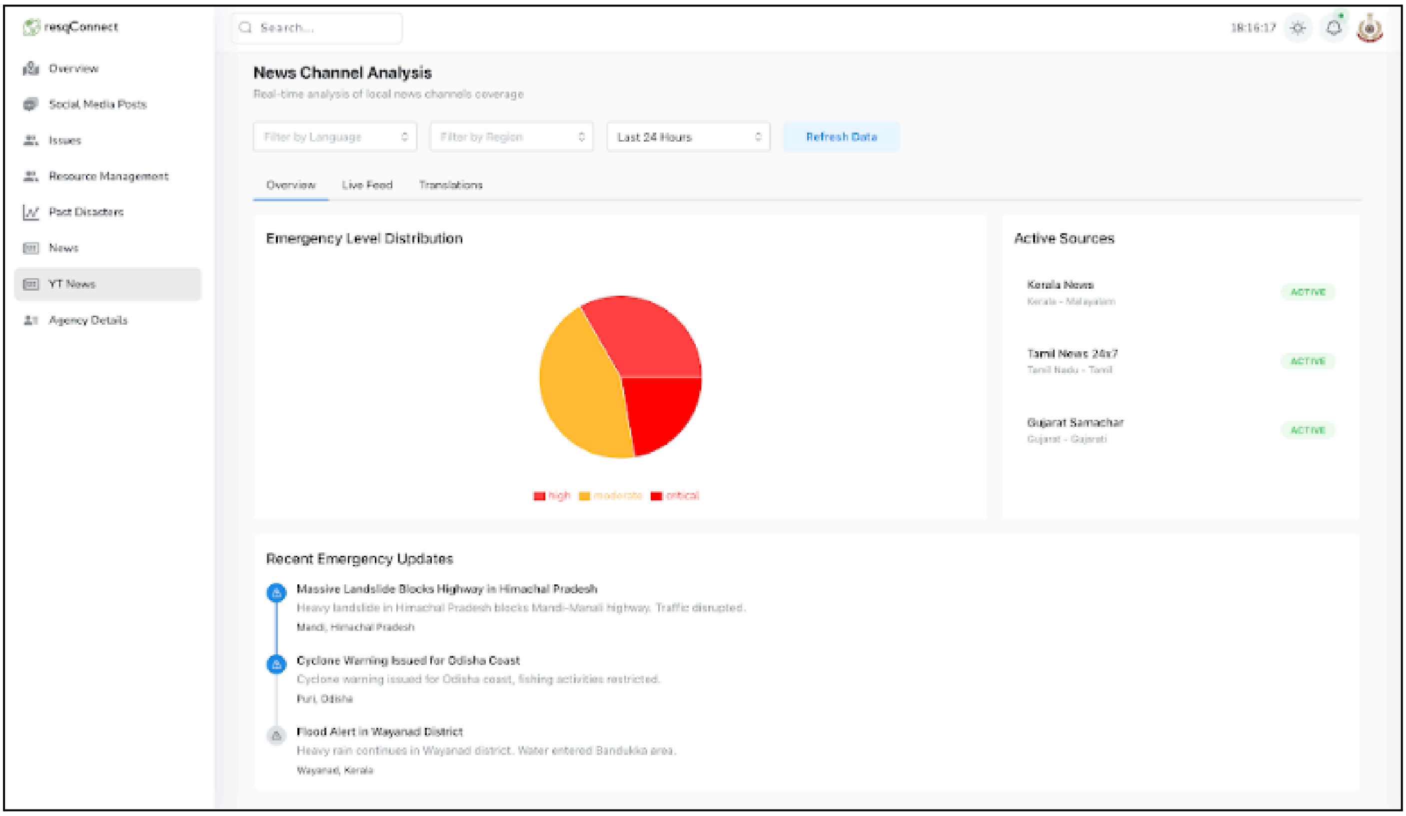

Figure 7.1.8 UI Screenshot Real-Time Rescue Updates from News Sources

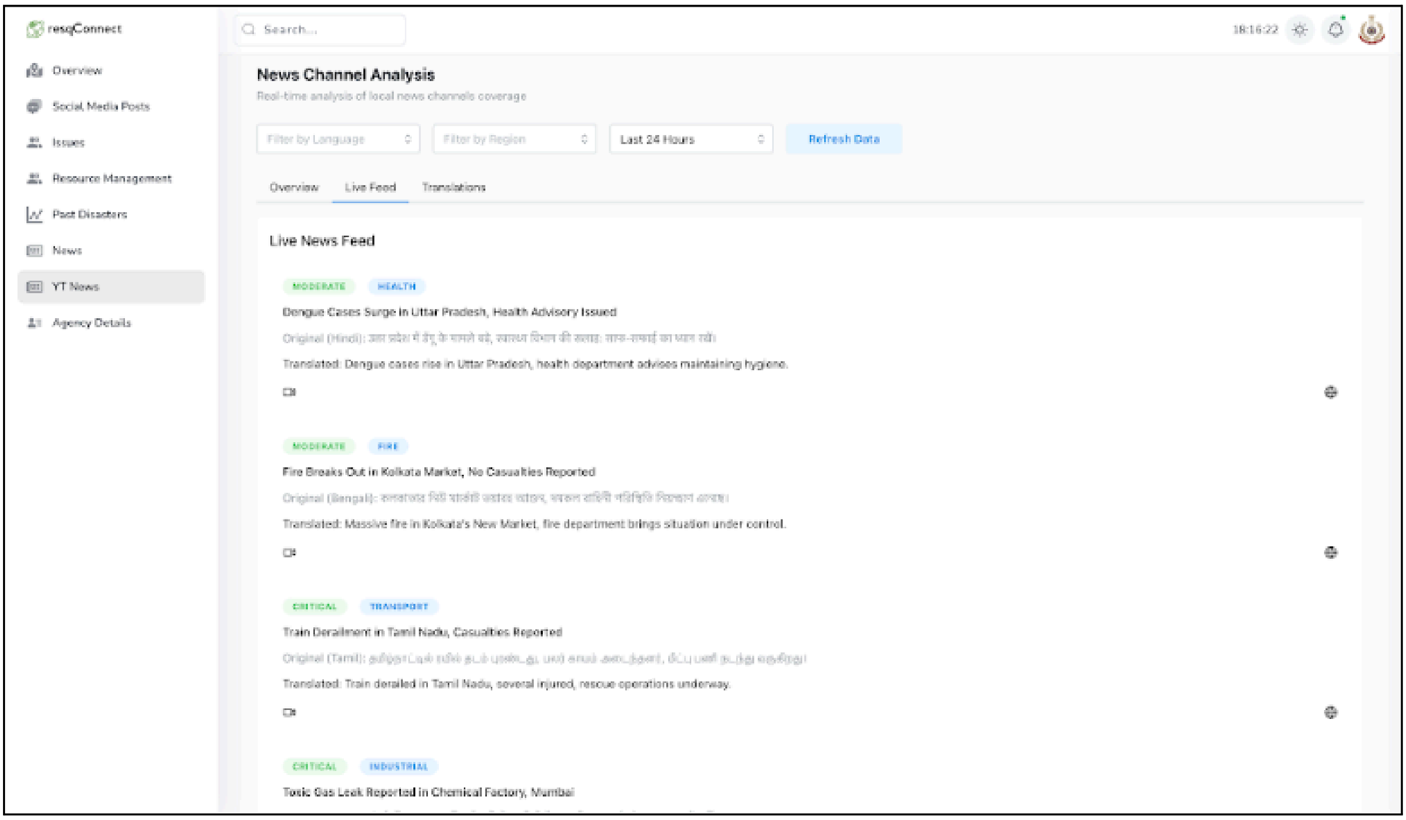

Figure 7.1.9 UI Screenshot Real-Time News Feed

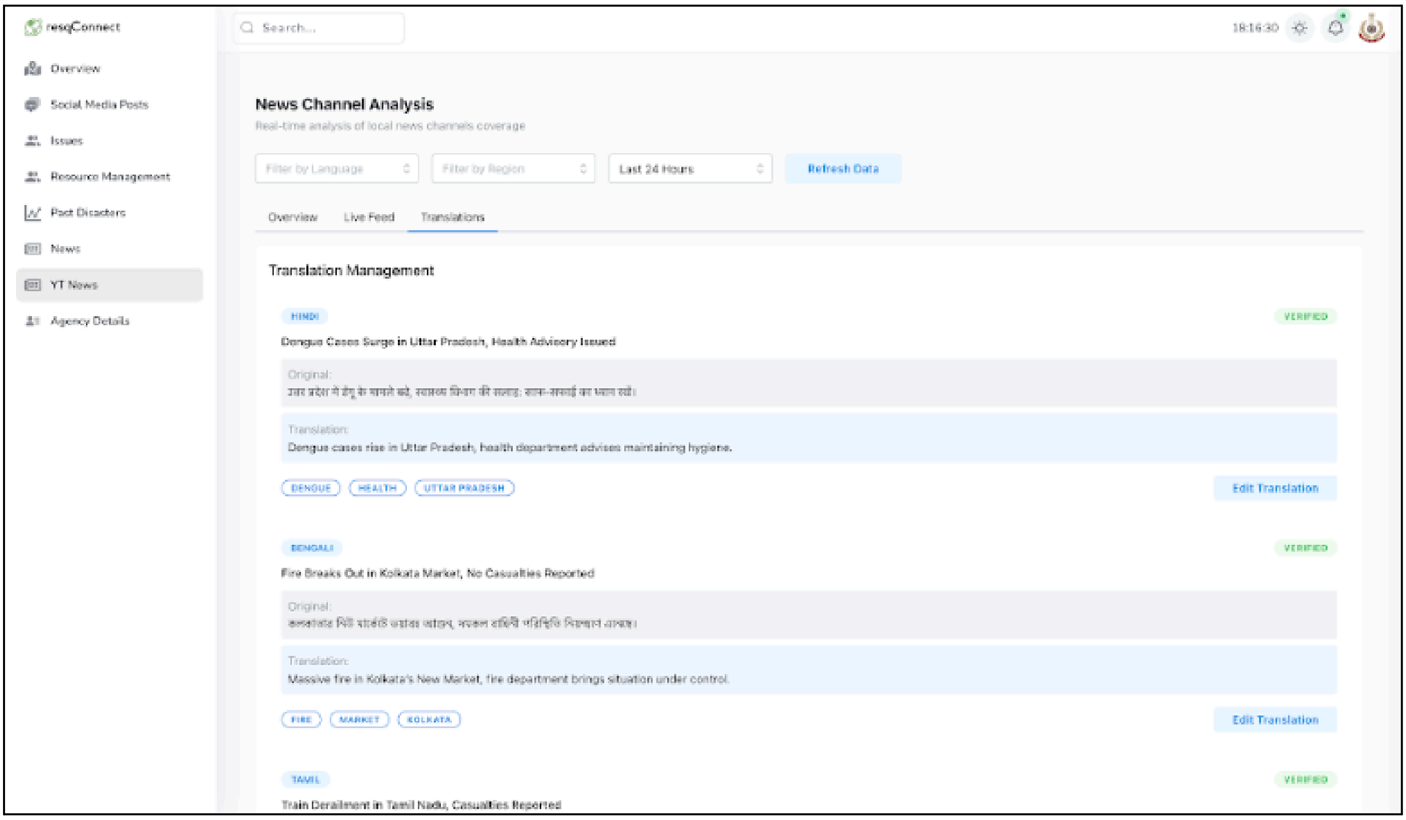

Figure 7.1.10 UI Screenshot Real-Time Local News Data Translated to English

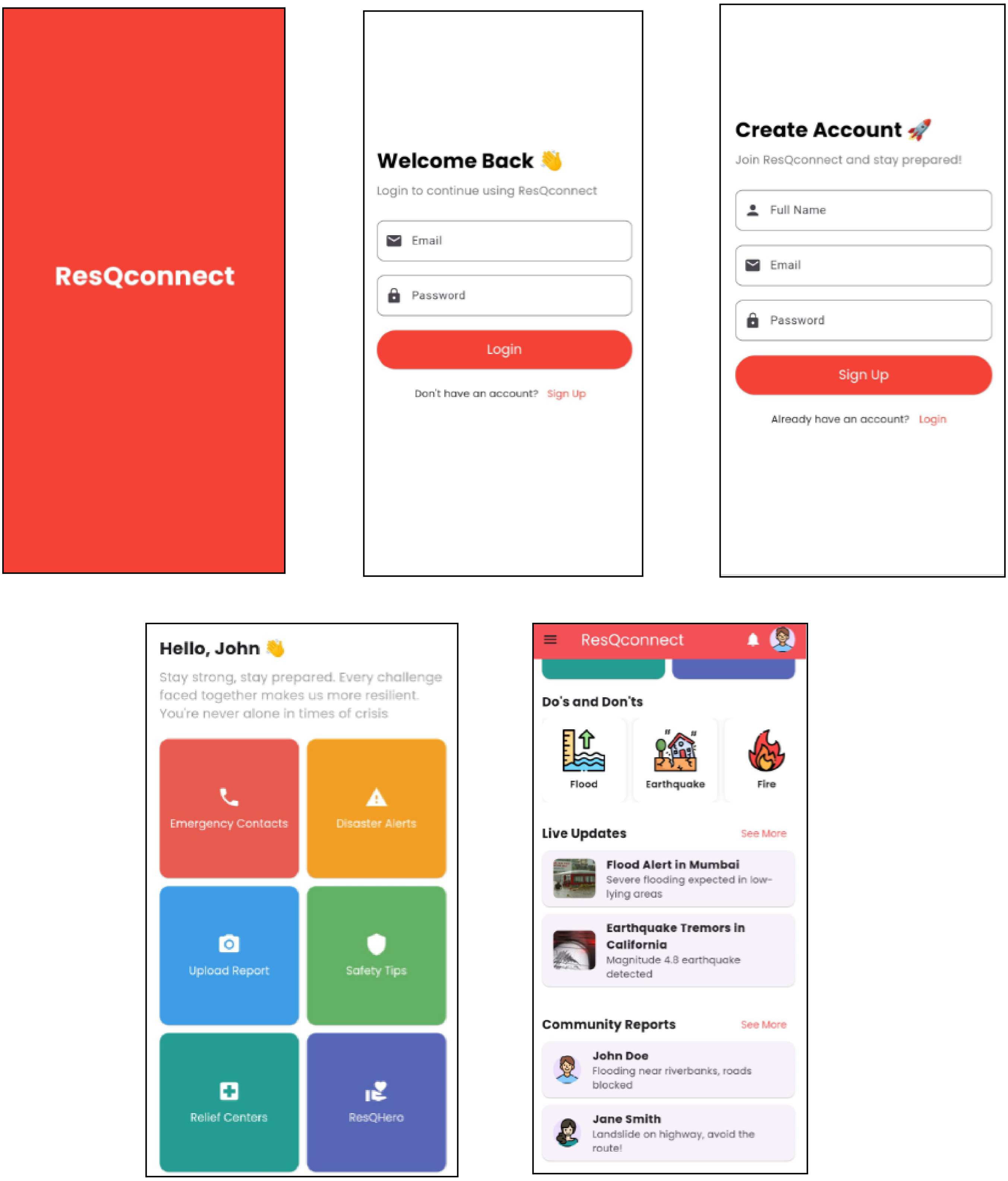

Figure 7.1.11 UI Screenshot of our App - User Authentication and Homepage Screens and Real-Time Rescue Updates

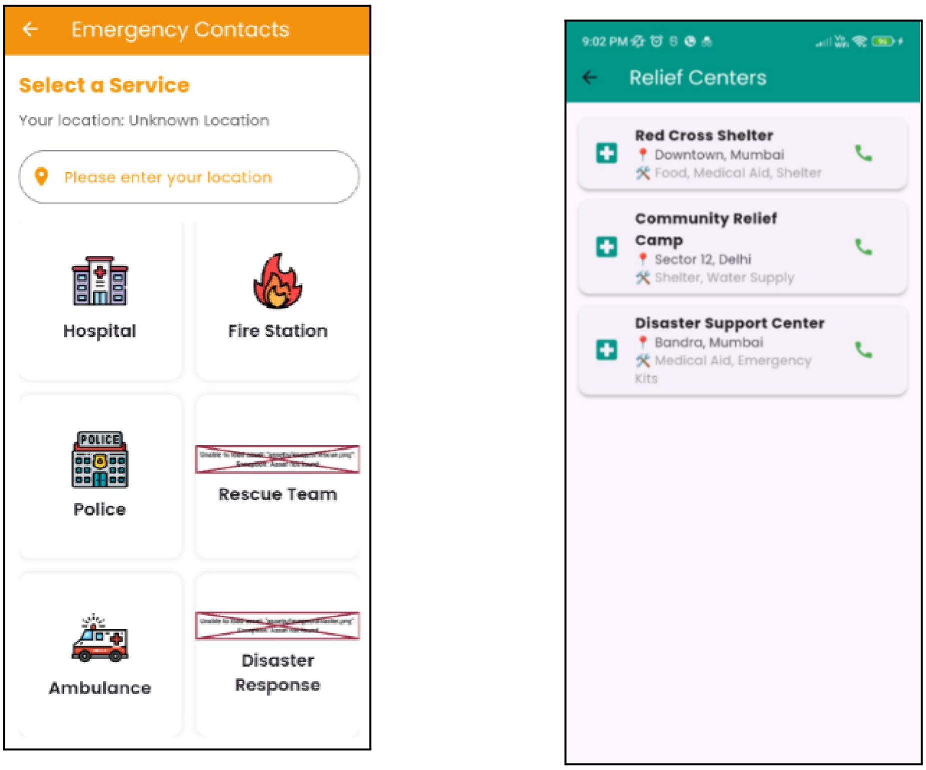

Figure 7.1.12 Emergency Contacts Module

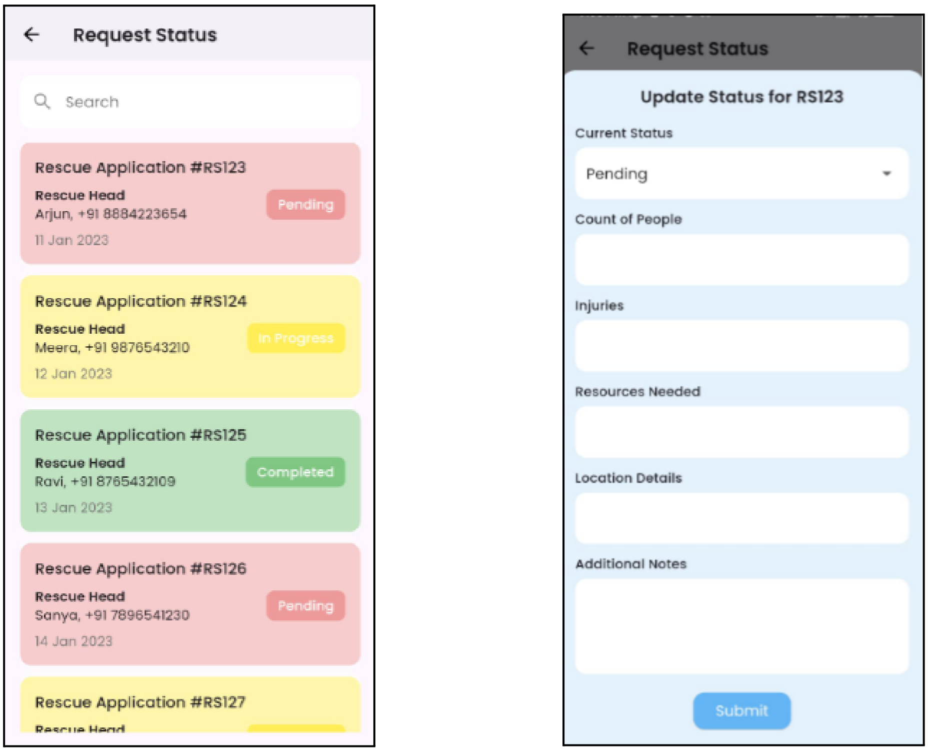

Figure 7.1.13 Rescue Status, Rescue Tracking, and Request Submission Screens

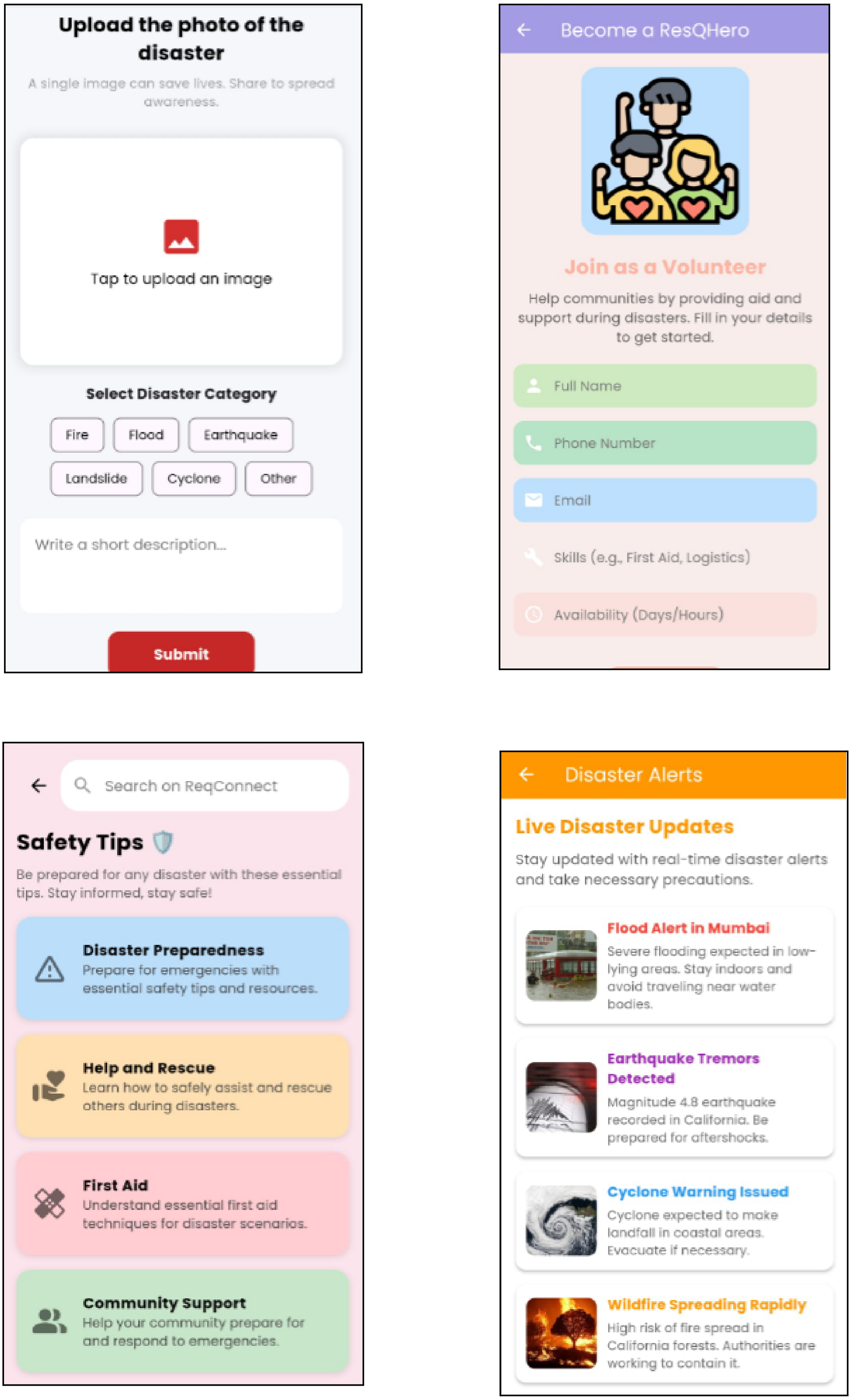

Figure 7.1.14 Upload Disaster Image, Volunteer Page, Alerts, and Safety Tips

7.2. Performance Evaluation measures

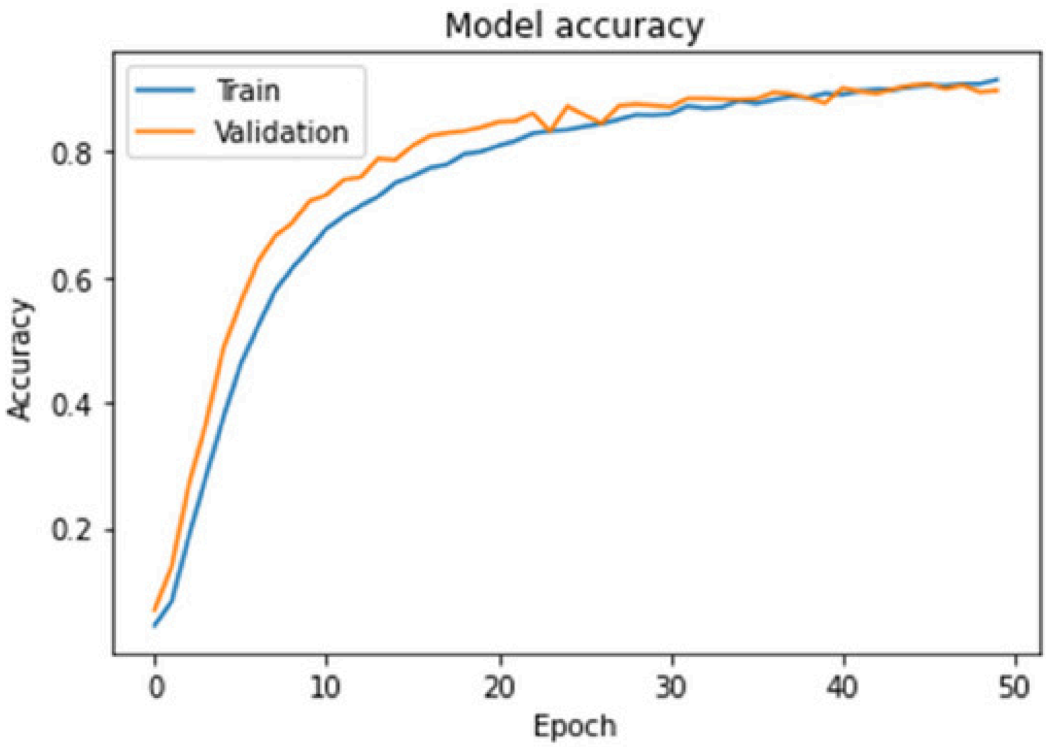

Figure 7.2.1 Train & Validation Accuracy Over Epochs

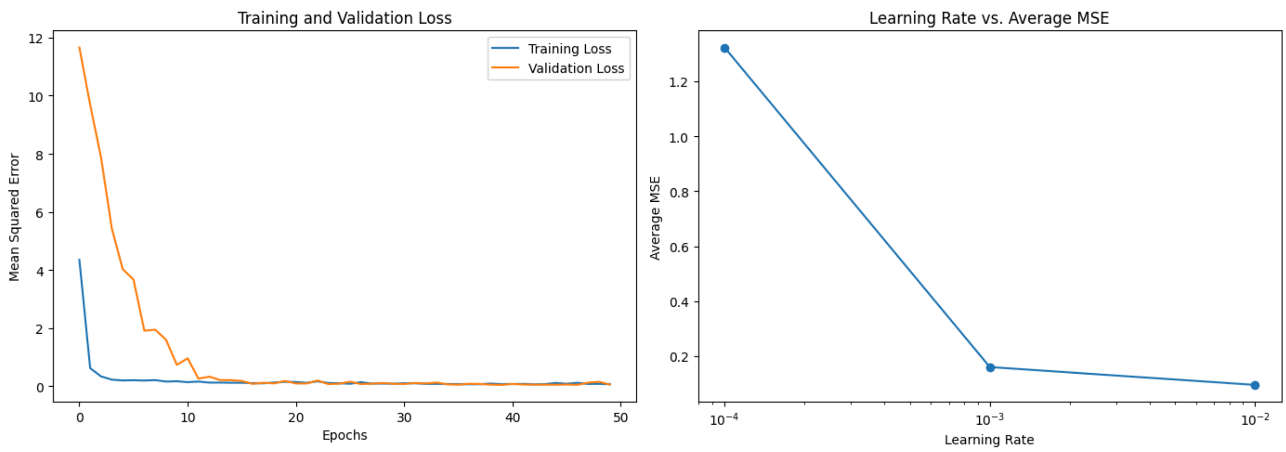

Figure 7.2.2 Train & Validation Loss Over Epochs and Learning Rate vs Average MSE

7.3. Input Parameters/Features considered

In this project, the input dataset primarily comprises **multimodal disaster data** obtained from **Twitter**, involving both **textual tweets** and **corresponding images**. Each sample in the dataset contains four key attributes:

Feature Name	Description
filename	The unique identifier for each text entry corresponding to the disaster tweet.
tweet	The textual content extracted from the Twitter post, often containing disaster descriptions, hashtags, and public reactions.
label	The ground truth class associated with the data point (e.g., `flood`, `fire`, `non_damage`), indicating the nature and severity of the event.
image	The associated disaster image scraped alongside the tweet, providing visual context for the incident described in the text.

Table 7.3.1 Data and Features Description from Twitter Data

The dataset reflects real-world noise, informal language, and varied quality in both text and image modalities, accurately simulating the challenges of real-time disaster monitoring.

Given the objective to achieve robust classification across different disaster types, additional data streams from structured news articles were incorporated. News articles were retrieved using APIs like NewsAPI and SerpAPI, and parsed into the following parameters:

Feature Name	Description
title	Headline of the news article.
description	Summary or introductory paragraph of the article.
published date	Timestamp indicating the publication time of the article.
source	The news agency or website providing the article.
url to image	Associated image URL where available, used for visual analysis.

Table 7.3.2 Data and Features Description from News Data

Preprocessing Steps for both tweet and news data included:

- **Text:** Tokenization, lowercasing, stop-word removal, padding/truncation, and embedding using BERT and XLNet models.

- **Images:** Resizing, normalization, data augmentation (flip, rotation), and noise reduction before passing to CNN architectures such as EfficientNetB3, DenseNet201, and ResNet50.

Feature Fusion Strategy:

In the final multimodal classification architecture, text embeddings (from tweets and news descriptions) and image feature maps were fused to generate a richer and context-aware representation of each disaster event. This fusion greatly improved disaster categorization accuracy by leveraging complementary strengths of text and image data.

The thoughtful selection and preprocessing of these features enabled the system to generalize well across multiple disaster scenarios, even in noisy, multilingual, and visually ambiguous contexts often encountered during real-world disaster events.

7.4. Graphical and statistical output

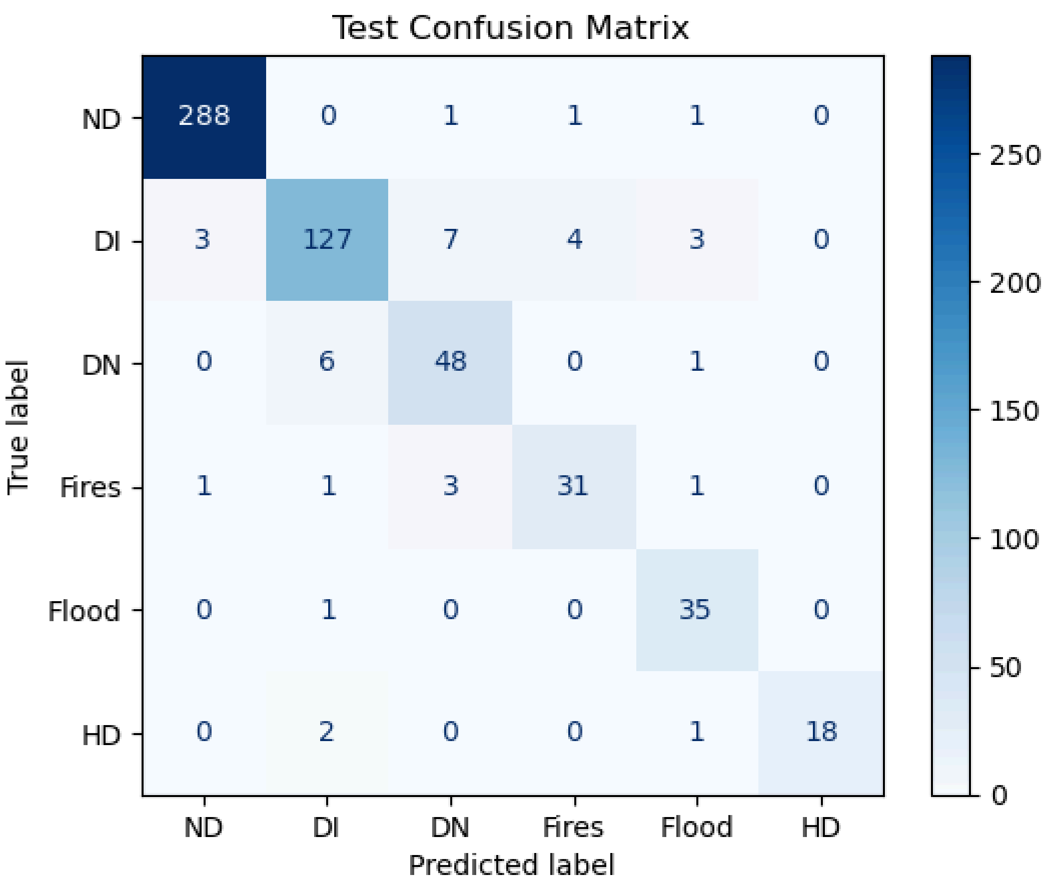

Figure 7.4.1 Confusion Matrix

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

            non_damage       0.99      0.99      0.99       291
damaged_infrastructure       0.93      0.88      0.90       144
        damaged_nature       0.81      0.87      0.84        55
                 fires       0.86      0.84      0.85        37
                 flood       0.83      0.97      0.90        36
          human_damage       1.00      0.86      0.92        21

              accuracy                           0.94       584
             macro avg       0.90      0.90      0.90       584
          weighted avg       0.94      0.94      0.94       584
```

Figure 7.4.2 Classification Report

7.5. Comparison of Results with Existing Systems

Existing Systems	Their Purpose	Limitations	Our Solution
1. NDMA	To formulate policies, plans, and guidelines for disaster management at the national level.	Focused on policy and planning; lacks real-time updates during disasters, which hinders immediate response.	Data collection from multiple data sources and using real time data using API integrations.
2. dISHA	Provides information and resources for disaster preparedness and response.	Data not frequently updated; relies on user-generated content, which vary in accuracy and reliability.	User uploaded content is checked for its reliability and severity and then considered for further insights.
3. Prutech	A mobile app designed for emergency response coordination and information dissemination.	Limited geographic coverage; does not include all types of disasters or provide comprehensive local resources.	Application for users to post photos thus covering all types of disasters
4. Suraksha App	Helps users report emergencies and receive alerts during disasters.	Requires internet connectivity, which is unreliable in affected areas; can be ineffective in remote locations.	Stores data in local storage and syncs with the backend on getting the internet connection

5. Sahana	Provides open-source software solutions for disaster management and humanitarian assistance.	Requires technical expertise for setup and implementation; not widely adopted across all regions.	No need of experts for setting up , it is user friendly and open sourced.
6. IMD	Provides weather forecasts, warnings, and updates related to meteorological disasters.	Primarily focused on weather-related events; does not cover other disaster types.	Supports all disaster types. Sends real-time, area-specific alerts. Automated Notifications

Table 7.5.1 Comparison of Results with Existing Systems

7.6. <u>Inference Drawn</u>

The experimental results and extensive performance evaluations substantiate the effectiveness of our proposed multimodal disaster classification framework. The integration of text and image modalities demonstrated significant gains in classification accuracy, precision, recall, and F1-score compared to unimodal baselines. Specifically, the fusion of features extracted through EfficientNetB3 for visual inputs and BERT/XLNet for textual representations enabled the system to capture both semantic and contextual nuances inherent in disaster scenarios. The system exhibited high resilience in handling noisy, multilingual, and incomplete data, a critical advantage for real-world deployments where data quality is highly variable. Furthermore, the real-time ingestion and processing pipelines, validated through end-to-end testing and live data simulations, showcased the system's scalability and robustness under dynamic conditions. The performance metrics affirm that our approach not only enhances situational awareness for rescue agencies but also provides a reliable decision-support mechanism in time-sensitive disaster response environments. These findings collectively validate the architectural choices and underline the pivotal role of multimodal deep learning in advancing next-generation disaster management systems.

Chapter 8 : Conclusion

8.1. Limitations

While ResQConnect integrates cutting-edge technologies like AI, real-time communication, and geolocation services, certain limitations persist. The system's functionality is dependent on stable internet connectivity, which can be disrupted during large-scale disasters. Additionally, the current version supports only English, which may restrict accessibility among non-English-speaking users, particularly in linguistically diverse regions.

1. **Internet Dependency:** Requires stable connectivity, which may not be available during severe disasters.
2. **Language Support:** Currently limited to English, reducing accessibility in multilingual areas.
3. **Scalability:** May face performance issues under extreme user load without advanced optimization.
4. **Misinformation Risk:** Social media data may include unreliable or misleading content.
5. **Device Access:** Assumes users have smartphones with GPS and sufficient battery, which may not always be the case.

8.2. Conclusion

ResQConnect was conceptualized with the core vision of revolutionizing disaster management through technological innovation. The project began with an extensive requirement-gathering phase, carefully studying existing gaps in emergency communication and data-driven response systems. Building on this foundation, we sourced multimodal datasets — primarily disaster-related tweets and images — through custom scraping pipelines and integrated additional structured news data from reputable APIs. This strategic data collection ensured that our models were exposed to real-world, noisy, and multilingual disaster scenarios.

The implementation phase saw the deployment of state-of-the-art AI models such as EfficientNetB3 for image feature extraction and BERT/XLNet for textual understanding. A sophisticated fusion mechanism was designed to combine these modalities, enabling rich context-aware disaster classification. The project culminated in the seamless integration of a user-centric mobile app and a dynamic, real-time admin dashboard — both interconnected via WebSockets for instantaneous updates.

What makes ResQConnect stand unique is its holistic and end-to-end approach: from automated real-time disaster detection to coordinated communication between affected citizens, rescue teams, and administrators. Unlike traditional systems that rely solely on static alerts or manual inputs, ResQConnect dynamically fuses real-time social media content, news feeds, and user reports into actionable intelligence. Its pastel-themed, intuitive design ensures accessibility even under high-stress conditions, transforming smartphones into critical life-saving devices. In essence, ResQConnect not only responds to emergencies but proactively builds resilience, positioning itself as a pivotal asset in the modern emergency management landscape.

8.3. Future Scope

ResQConnect lays a strong foundation for intelligent disaster response, yet several avenues remain open for enhancement to increase its real-world impact and scalability.

Multilingual Support:
Add regional language options using models like mBERT to improve accessibility across India.

Drone & Satellite Integration:
Use aerial imagery for real-time surveillance and automated hazard detection in remote areas.

Offline Functionality:
Enable offline access with delayed data syncing to ensure usability during network outages.

Crowdsourced Reporting:
Allow users to upload live images/videos to enhance situational awareness and model retraining.

Government Integration:
Connect with NDMA and state agencies via APIs for coordinated and verified disaster response.

Predictive Analytics:
Use historical and live data for forecasting disasters and improving preparedness.

Accessibility Enhancements:
Add voice support and UI adjustments for differently-abled users to ensure inclusivity.

Through these enhancements, ResQConnect has the potential to evolve into a comprehensive, nationally scalable disaster management platform that not only responds to ongoing crises but also anticipates and mitigates their impact through data-driven intelligence.

References

[1] Wiegmann, M., Kersten, J., Senaratne, H., Potthast, M., Klan, F., and Stein, B., 2021, "Opportunities and Risks of Disaster Data from Social Media: A Systematic Review of Incident Information," *Natural Hazards and Earth System Sciences*, 21, pp. 1431–1450. DOI: 10.5194/nhess-21-1431-2021.

[2] Amitangshu Pal, Wang, J., Wu, Y., Kant, K., Liu, Z., and Sato, K., 2023, "Social Media Driven Big Data Analysis for Disaster Situation Awareness: A Tutorial," *IEEE Transactions on Big Data*, 9(1), pp. 24–39. DOI: 10.1109/TBDATA.2022.3158431.

[3] Hao, H., and Wang, Y., 2020, "Leveraging Multimodal Social Media Data for Rapid Disaster Damage Assessment," *International Journal of Disaster Risk Reduction*, 51, pp. 101760. DOI: 10.1016/j.ijdrr.2020.101760.

[4] You, J., Lee, K., and Kwon, H.-Y., 2024, "DeepScraper: A Complete and Efficient Tweet Scraping Method Using Authenticated Multiprocessing," *Data & Knowledge Engineering*, 149, pp. 102260. DOI: 10.1016/j.datak.2023.102260.

[5] Mishra, S. S., Bisen, A., Mundhada, S., Singh, U., and Bongirwar, V., 2022, "DIVVA: Disaster Information Verification and Validation Application Using Machine Learning," *International Journal of Next-Generation Computing*, 13(5), pp. 123–130.

[6] Oprea, S. V., and Bâra, A., 2022, "Why Is More Efficient to Combine BeautifulSoup and Selenium in Scraping for Data Under Energy Crisis," *Ovidius University Annals, Economic Sciences Series*, Vol. XXII, Issue 2, pp. 146–152.

[7] Ningsih, A. K., and Hadiana, A. I., 2021, "Disaster Tweets Classification in Disaster Response Using Bidirectional Encoder Representations from Transformer (BERT)," *IOP Conf. Ser.: Mater. Sci. Eng.*, 1115, 012032. DOI: 10.1088/1757-899X/1115/1/012032.

[8] Domala, J., et al., 2020, "Automated Identification of Disaster News for Crisis Management using Machine Learning and Natural Language Processing," in *Proc. 2020 Int. Conf. Electronics and Sustainable Communication Systems (ICESC)*, Coimbatore, India, pp. 503–508. DOI: 10.1109/ICESC48915.2020.9156031.

[9] Kusumasari, B., and Prabowo, N. P. A., 2020, "Scraping Social Media Data for Disaster Communication: How the Pattern of Twitter Users Affects Disasters in Asia and the Pacific," *Natural Hazards*, 103, pp. 2615–2635. https://doi.org/10.1007/s11069-020-04136-z.

[10] Laya, M., Delimayanti, M. K., Mardiyono, A., Setianingrum, F., Mahmudah, A., and Anggraini, D., 2021, "Classification of Natural Disaster on Online News Data Using Machine Learning," in *Proc. 2021 5th Int. Conf. Electrical, Telecommunication and Computer Engineering (ELTICOM)*, Medan, Indonesia, pp. 42–46. DOI: 10.1109/ELTICOM53303.2021.9590125.

[11] Aswathy, A., Rekha, P., Lakshmi, G., Divya, P., Maneesha, V., and Ramesh, V., 2022, "An Efficient Twitter Data Collection and Analytics Framework for Effective Disaster Management," in *Proc. DELCON*, DOI: 10.1109/DELCON54057.2022.9753627.

[12] Jayan, M., and Jacob, L., 2023, "Categorizing Disaster Tweets Using Learning-Based Models for Emergency Crisis Management," in *Proc. ICACCS*, DOI: 10.1109/ICACCS57279.2023.10113105.

[13] Gnanasekaran, P., 2023, "Juncture of Text Preprocessing Techniques & Extracting Sentiment Analyzing of Micro-Blog Based on Machine Learning Algorithms," in *Proc. ICSES*, DOI: 10.1109/icSES60034.2023.10465450.

[14] Mahajan, P., Raghuwanshi, P., Setia, H., and Randhawa, P., 2024, “A Multi-Model Approach for Disaster-Related Tweets,” *Journal of Computer Modelling and Methods*, 3(2), DOI: 10.57159/gadl.jcmm.3.2.240125.

[15] Singh, P., Singh, M., Das, P., and Chand, S., 2024, “Leveraging Domain-Adapted Transformer Based Models for Disaster-Specific Tweet Classification,” *J. Intell. Fuzzy Syst.*, DOI: 10.3233/jifs-219419.

[16] Chen, W., and Fang, J., 2024, “Optimizing AI-Driven Disaster Management through LLMs,” *Preprints*, DOI: 10.20944/preprints202407.1446.v1.

[17] Frankie, Y. A. S., 2023, “Implementation of Text Indexing System in Web-Based Document Search Application Using MongoDB,” *Jurnal Teknik Informatika*, 4(5), DOI: 10.52436/1.jutif.2023.4.5.959.

[18] Dasari, S. K., Srinivas, G., and Reddy, P. J., 2024, “A Deep Parallel Hybrid Fusion Model for Disaster Tweet Classification on Twitter Data,” *Decision Analytics Journal*, DOI: 10.1016/j.dajour.2024.100453.

Patents

[1] Y. Chia-Hung, “Context-aware social media disaster response and analysis,” U.S. Patent 9,408,051 B2, Aug. 2, 2016.

[2] D. R. Witbrock and M. D. Witbrock, “Social media analytics for emergency management,” U.S. Patent 20200126174 A1, Apr. 23, 2020.

[3] A. Nagpal, S. Goel, and J. Dube, “Social media content for emergency communication,” U.S. Patent 20190174289 A1, June 6, 2019.

Appendix

I. Paper 1 details:

1. <u>Paper Published</u>

Our paper has been published in:

- **Journal Name:** Communications on Applied Nonlinear Analysis
- **ISSN:** 1074-133X
- Scopus Indexed Journal

Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 3s (2025)

AI-Powered Multimodal Disaster Response Enhancement using Social Media Streams

Rohini Temkar[1], Gresha Bhatia[2], Sai Thikekar[3], Arya Banavali[4], Yash Chhaproo[5], Aradhya Ingle[6]

[1,2,3,4,5,6]Department of Computer Engineering, Vivekanand Education Society's Institute of Technology, Mumbai, India

Article History:

Received: *26-09-2024*

Revised: *04-11-2024*

Accepted: *19-11-2024*

Abstract:
Social media serves as a valuable source of real-time disaster data, providing timely updates that are critical for effective disaster monitoring and response. It plays an essential role in raising awareness, enhancing preparedness, and facilitating prompt action during emergencies. While much of the current research focuses on disaster prediction, real-time updates, which are key to saving lives and coordinating response efforts, are often overlooked. Our project addresses this gap by focusing on the collection and analysis of real-time data from various social media platforms, including disaster-related images and textual data like tweets.We use multiple APIs to gather real-time data from platforms such as Instagram, Facebook, and Twitter. To analyze visual data, we employ state-of-the-art multimodal models like ResNet50 and EfficientNet for image classification. For the textual data, advanced natural language processing models like BERT and XLNet are used for sentiment analysis, event detection, and severity assessment. By integrating and fusing the image and text data, our system is able to identify the type of disaster, estimate its severity, and provide actionable insights.

Keywords: Real-time disaster data, APIs, multimodal, ResNet50, EfficientNet, BERT, XLNet , disaster severity assessment.

I. INTRODUCTION

Natural disasters, including tsunamis, floods, and earthquakes, are destructive and often beyond human control. However, disaster management offers a structured approach to mitigate the devastating impacts of such events through preparedness, response, and recovery strategies. India, in particular, ranks third globally in human mortality from extreme weather events between 2000 and 2019. Over the period from 1983 to 2011, the country faced 190 floods and 54 cyclones, leading to severe human and economic losses, with approximately 60,919 deaths and 1.22 billion individuals affected by floods, and 20,360 deaths and 56 million people impacted by cyclones. These statistics emphasize the critical vulnerability of India to natural disasters and the urgent need for more effective disaster management and resilient infrastructure systems to reduce future risks [1].

In recent years, social media platforms, such as Twitter, Instagram, and Facebook, have played an increasingly important role in disaster management by providing real-time data that enhances situational awareness and facilitates rapid emergency response. The integration of social media analytics into disaster management frameworks has revolutionized how information is collected, processed, and acted upon during crises. Social media's ability to provide up-to-date, location-specific data enables authorities and responders to monitor dynamic situations and coordinate more efficient responses [2].

Further, multimodal data—text, images, and videos—has proven essential for enhancing the precision of urban event monitoring. For instance, [2] and related studies demonstrate that combining various data types significantly improves the spatial accuracy of event detection, such as urban flooding, achieving up to 100% precision in specific instances ("A Spacel Information Extraction Method based on Multimodal Social Media Data: A Case Study on Urban Inundation," 2023). Moreover, Chatterjee [3] propose an innovative approach for monitoring mental health through social media posts, using sentiment analysis and multimodal features to construct a mental health index, achieving 89% accuracy with Support Vector Machines (SVM). This real-time analysis has profound implications for disaster response, as psychological well-being is a critical aspect of managing the human impact of disasters.

Disaster response can also be enhanced by employing advanced natural language processing models like Late Dirichlet Allocation and BERT. [2] show that real-time classification of disaster-related topics using such models improves accuracy by 12% compared to earlier techniques. Despite these advancements, challenges remain, especially in managing the noisy and unstructured nature of social media data, which can affect the reliability of insights. Ensuring a balance between high accuracy and the unpredictability of social media remains a vital area for ongoing research.

II. RELATED WORK

Efficient data collection and aggregation from diverse sources, such as social media and news outlets, play a crucial role in real-time disaster management systems. In recent work, DeepScraper by Jaebeom [4] demonstrated a novel approach to tweet scraping using authenticated multiprocessing, significantly improving the collection speed and volume of social media data. This method successfully scraped over 5 million tweets, outperforming the standard Twitter API by 23.7 times, making it highly applicable for the social media data collection component of disaster response systems.

To complement social media data collection, the aggregation of disaster-related news is essential for a comprehensive crisis management system. Domala [5] proposed a system that automates the collection of disaster news using web scraping techniques and classifies the gathered data through machine learning models such as Support Vector Machines (SVM) and Logistic Regression. This system also incorporates Natural Language Processing (NLP) to preprocess the news data and geoparses relevant disaster locations. With an evaluation based on precision, recall, F1 score, and a confusion matrix, the approach ensures that only relevant news is included in crisis management workflows.

Accurate data validation is critical in disaster management to prevent the spread of misinformation, especially on social media platforms. Mishra [6] addressed this challenge through the Disaster Information Verification and Validation Application (DIVVA), which uses machine learning to verify disaster-related information by cross-referencing it with official government sources. The system utilizes a Bidirectional LSTM model to classify tweets as real or fake based on textual input analysis, achieving an accuracy of 84%. DIVVA's approach to ensuring the reliability of disaster information is highly relevant to our project, which focuses on aggregating and validating data from social media and news, ensuring only verified and actionable information is included in disaster response efforts.

Beyond data collection and validation, effective categorization of disaster-related information is essential for timely response. The work by Mishra [6] introduced a deep multimodal learning approach that integrates visual and textual features to classify disaster types from social media posts. The system employs ResNet50 for visual feature extraction and a BiLSTM model with an attention mechanism for textual analysis, showing a significant performance boost over unimodal models and even improving by 7% compared to other multimodal approaches. This method addresses the challenge of processing vast amounts of social media data, ensuring more accurate disaster categorization. Incorporating such a multimodal approach into our project can enhance the classification of disaster data, making the system more robust in identifying and organizing real-time information.

Effective classification of disaster-related tweets is crucial for timely response and rescue efforts. Ningsih [7] explored the use of the Bidirectional Encoder Representations from Transformers (BERT) model to enhance the identification of disaster-related tweets, particularly focusing on sentiment analysis. This approach helps differentiate critical tweets, such as rescue requests, from general disaster discussions, addressing the challenge of understanding complex language structures during emergencies. BERT's ability to manage context and sentiment uncertainty proves beneficial in improving tweet classification, thereby aiding disaster response teams in prioritizing rescue operations. Integrating BERT-like models into our project can significantly enhance the accuracy and relevance of disaster data classification.

III. PROPOSED METHODOLOGY

In natural disasters, platforms like Twitter become critical for real-time updates. The proposed solution addresses the challenge of collecting, processing, and classifying disaster-related tweets, overcoming Twitter's API restrictions and large data volumes. The system integrates web scraping, real-time streaming, and machine learning to efficiently filter and classify tweets, thereby improving disaster response.

3.1. Data Collection via Web Scraping

Due to Twitter's recent API restrictions, web scraping has become an essential alternative for collecting tweets. This solution leverages Selenium and BeautifulSoup to scrape tweet content, metadata, and user information directly from Twitter's web interface. To handle Twitter's rate limits and prevent detection, multiple techniques are employed. One such technique is proxy rotation, which uses a rotating pool of proxies to prevent IP bans and ensure consistent scraping throughout the process [8]. Additionally, automated scrolling is simulated using Selenium to load and extract older tweets that are beyond the initial page view [9].

Furthermore, throttling and user-agent spoofing are implemented to mimic human behavior and reduce the risk of being detected as a bot. By introducing randomized delays between requests and changing user-agents at regular intervals, the system closely emulates typical browsing patterns, allowing it to bypass Twitter's anti-bot mechanisms [10]. These combined methods ensure scalable and efficient tweet collection despite the recent restrictions imposed by Twitter's API, enabling a continuous stream of relevant data for disaster-related analysis.

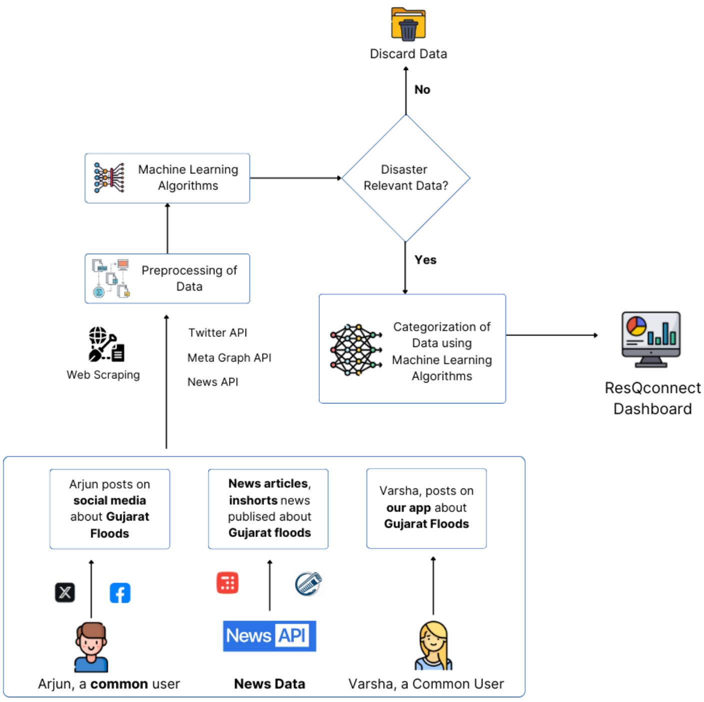

Fig 1. Methodolgy

Figure 1 illustrates the process flow of collecting data via web scraping and real-time streaming, followed by categorizing tweets using machine learning models. This flow starts from the data scraping process, streaming to Kafka topics, followed by NLP processing, classification, and storage.

3.2. Real-Time Data Ingestion and Processing Pipeline

To handle the influx of disaster-related tweets, a robust real-time data ingestion and processing pipeline is established using Apache Kafka and Apache Spark. This pipeline is critical for ensuring that the system can manage the large volume of tweets in real-time, particularly during high-traffic disaster periods.

3.3 Kafka-Based Data Ingestion

Tweets are streamed into Kafka topics, where they are tagged with disaster-related metadata such as "earthquake" or "flood." Kafka’s partitioning capability plays a significant role in this process, allowing the system to divide and handle tweets in parallel based on metadata, such as disaster type or geographic location. This enables the system to efficiently manage multiple streams of disaster-related data without bottlenecks [11]. In practice, the partitioning ensures that different disaster events are processed simultaneously, offering real-time insights on various ongoing disasters. Each Kafka topic serves as a queue for a particular category, which guarantees that no tweet is lost or delayed, even when traffic spikes. The combination of Kafka's durability and parallelism ensures a continuous flow of data into the processing system, ready for real-time analytics.

To avoid reliance on Twitter's paid API, we leveraged twscrape, a web-scraping tool that efficiently extracts tweets based on specific disaster-related queries. The extracted tweets are then streamed into Apache Kafka in real time, where each tweet is stored in a structured JSON format. The key code snippets for our pipeline are as follows:

Producer Code:

```
from kafka import KafkaProducer
import twscrape
import asyncio

# Async function to scrape tweets and send to Kafka
async def fetch_and_send_tweets(query, kafka_topic):
    scraper = twscrape.TweetScraper(query)
    async for tweet in scraper.get_items():
        tweet_data = {'id': tweet.id, 'text': tweet.content,
                      'timestamp': tweet.date}
        producer.send(kafka_topic, value=tweet_data)
```

Consumer Code:

Here, tweets are scraped asynchronously and sent to the disaster_tweets topic in Kafka for further processing.

```

from kafka import KafkaConsumer
import json

# Function to process tweets from Kafka
def process_tweets():
    for message in consumer:
        tweet = message.value
        print(f"Processing Tweet: {tweet['text']}
              from user: {tweet['username']}")

```

This consumer continuously listens for new tweets, processing them in real time, which is essential for tasks like disaster response monitoring.

3.4 Real-Time Processing with Spark Streaming

Once the tweets are ingested into Kafka, they are passed to Apache Spark Streaming, which processes the data in micro-batches for real-time analytics. Key tasks performed during this stage include sentiment analysis, where text mining techniques are applied to assess the sentiment of tweets. This provides immediate insight into public reactions and emotions concerning specific disasters [12]. Additionally, Spark's NLP tools are used to extract key disaster-related keywords and location data from the tweets, helping in the identification of affected regions and relevant discussions. The use of Spark ensures that the system can scale to handle massive amounts of tweet data efficiently, even during disaster events with high tweet volumes. By leveraging its distributed processing capabilities, the system guarantees that the data is processed quickly and effectively, making the insights available in real-time. This approach ensures fast, scalable, and reliable processing, which is essential during critical disaster scenarios.

3.5 Multimodal Model for Tweet Categorization

Disaster-related tweets often contain both textual and visual information, making a multimodal approach necessary for accurate categorization. To handle this, the system employs natural language processing (NLP) models for text and convolutional neural networks (CNNs) for images. The integration of these two modalities improves the classification of disaster-related tweets.

3.5.1 Text Processing using NLP Models

For text classification, BERT (Bidirectional Encoder Representations from Transformers) and XLNet models are fine-tuned on a dataset of disaster-related tweets. These models excel at contextual understanding, which is essential for accurately identifying disaster-related information within short, often informal tweets. BERT, in particular, considers the context of words bidirectionally, which allows for a more nuanced understanding of the text. This bidirectional context improves the model's ability to classify tweets with higher accuracy [13]. XLNet, on the other hand, is trained using a permutation-based language modeling approach, which allows it to handle longer sequences more effectively. This makes XLNet particularly suited for longer tweets or more complex sentence structures, where context and sequence are important for classification accuracy [14]. Both models, when fine-tuned, enhance the system's ability to differentiate disaster-related tweets from unrelated content based on their context.

3.5.2 Image Processing using CNNs

In addition to text, many disaster-related tweets include images that provide visual evidence of the event. For this, the system uses EfficientNetB3, DenseNet201, and ResNet50—all powerful CNN architectures trained on disaster imagery datasets. Among these, EfficientNetB3 demonstrated the best performance, striking a balance between accuracy and computational cost. This makes it particularly suitable for real-time image classification in disaster scenarios, where time is of the essence and resource efficiency is crucial [14]. DenseNet201 and ResNet50 also contribute to the model's ability to classify images by leveraging their depth and feature extraction capabilities. By using a combination of these CNNs, the system ensures that the visual data from tweets is processed effectively, providing additional context to the disaster categorization.

3.5.3 Multimodal Fusion Model Architecture

The integration of text and image data occurs in the multimodal fusion model, as depicted in Fig. 2. This internal flow diagram shows how features extracted from both modalities (text and image) are combined before the final classification. The fusion of both textual and visual features enables the model to more accurately detect disaster-related content. By leveraging both types of data, the system improves overall classification performance, ensuring that even tweets with limited text but strong visual content—or vice versa—are accurately categorized.

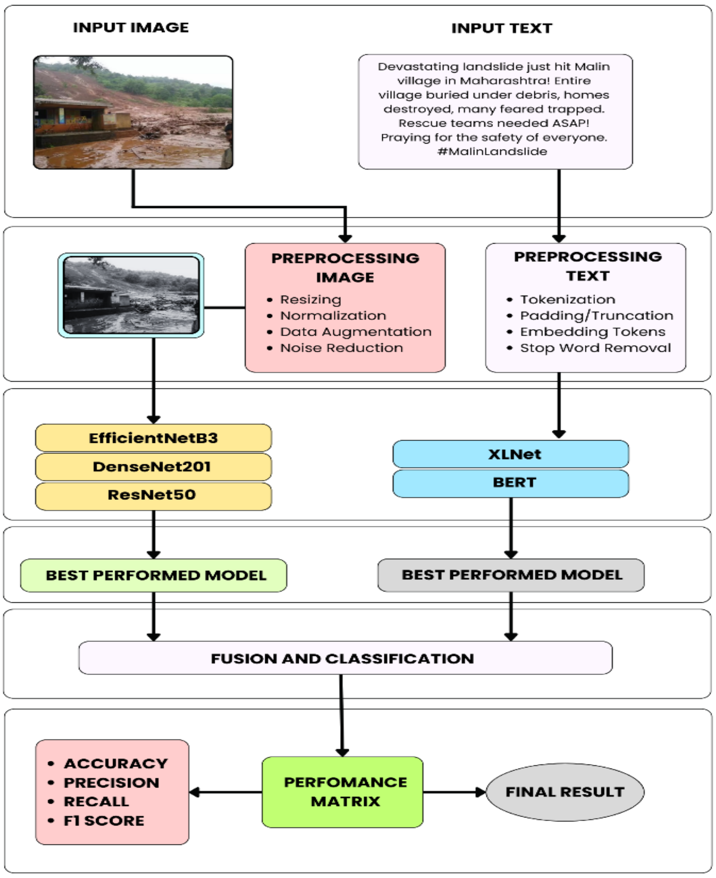

Fig 2. Multimodal Fusion Model Architecture

This multimodal approach provides a comprehensive solution to classifying disaster-related tweets, enhancing the reliability of detection by utilizing both language and imagery. 3. Fusion of Text and Image Data

The system enhances tweet classification by combining text and image features through a feature-level fusion approach. In this approach, the features extracted from both the textual data (using NLP models) and the visual data (using CNNs) are combined before being passed to the classifier. This fusion ensures that the model leverages the complementary strengths of both textual and visual data, leading to a more accurate classification of disaster-related tweets. Studies have shown that this multimodal fusion outperforms approaches that rely on a single modality, such as only text or only images, particularly in complex scenarios like disaster detection [9].

The fusion model is vital for handling tweets with mixed or incomplete data, where visual information or text alone may not be sufficient to classify the tweet accurately.

3.6. Filtering Irrelevant Tweets with Gemini LLM

To handle the vast influx of tweets, the system employs Gemini LLM, a large language model (LLM) designed specifically for filtering out irrelevant content. Tweets related to disasters can often be mixed with spam, advertisements, or other noise, which may not contribute meaningfully to disaster management efforts. Gemini LLM analyzes the context of tweets and effectively differentiates between disaster-relevant information and extraneous content. The model's ability to understand and assess tweet context in real-time significantly reduces the clutter in the dataset, ensuring that only meaningful tweets are processed and analyzed [15]. This filtering mechanism is crucial in maintaining the system's overall efficiency and focus during disaster events when accurate and relevant information is essential.

3.7. Data Storage and Search

After processing, the system stores tweets in MongoDB and utilizes Elasticsearch for indexing and searching. MongoDB is used for storing unstructured data, such as tweet content, images, and associated metadata, which makes it well-suited for handling diverse tweet formats. It ensures the scalability of the storage, as the volume of tweets can increase dramatically during disaster situations. In parallel, Elasticsearch provides the capability for fast, full-text search, enabling real-time querying and retrieval of tweets based on disaster-related keywords, geographic locations, or other metadata. This combination allows users to search the vast database quickly and efficiently. To visualize trends and provide insights into tweet patterns, the system also integrates Kibana, a data visualization tool that works seamlessly with Elasticsearch. Kibana allows real-time monitoring of tweet trends, helping disaster response teams make data-driven decisions based on the most up-to-date information [16].

3.8. Performance Evaluation

The system's performance is evaluated using several key metrics to ensure the effectiveness of tweet classification and processing. Accuracy is the primary metric used to assess the overall proportion of correctly classified tweets. However, given the imbalance in datasets—where some disaster types may be more frequent than others—additional metrics like precision, recall, and the F1-score provide a more granular view of performance. Precision measures the proportion of correctly identified disaster tweets out of all tweets classified as disaster-related, while recall evaluates the system's ability to identify all relevant disaster tweets. The F1-score, which is the harmonic mean of precision and recall, provides a balanced evaluation, particularly in cases of imbalanced datasets [17]. Finally, the system also monitors latency, measuring the time between tweet ingestion and classification. This ensures that the system can handle real-time tweet processing, an essential feature during fast-paced disaster events.

These combined efforts ensure that the system is both accurate and responsive, capable of processing large amounts of tweet data efficiently during critical times, and providing actionable insights to disaster response teams.

IV. IMPLEMENTATION

The proposed solution integrates various components for real-time disaster tweet collection, processing, and classification. The system is built using web scraping tools, Apache Kafka, Apache Spark, MongoDB, Elasticsearch, and a multimodal classification model that leverages BERT, EfficientNetB3, and other state-of-the-art machine learning tools.

4.1 Dataset

The dataset used in this research consists of social media posts with associated text and image files. Each post is classified into one of several categories, including damage-related events like floods or fires and non-damage events. The dataset structure looks like this:

Filename	Tweet	Label	Image
ad_2017-11-25_10-36-26.txt	⭐ We are really getting into the Christmas spirit...	non_damage	ad_2017-11-25_10-36-26.JPG
building_2017-10-30_17-26-34.txt	IJOY uv board has a competitive price and very...	non_damage	building_2017-10-30_17-26-34.JPG
floodwater_2017-09-04_04-46-10.txt	Arriving in Kalkundi island destroyed in #bangladesh...	flood	floodwater_2017-09-04_04-46-10.JPG
accrafloods_2015-06-06_16-59-56.txt	Hi my lovelies, check out my firsthand experience of...	flood	accrafloods_2015-06-06_16-59-56.JPG
buildingfire_2016-10-02_03-07-17.txt	The Hamilton fire service during an exercise at...	fires	buildingfire_2016-10-02_03-07-17.JPG

Table 1. Dataset overview

The dataset consists of columns for the filename, tweet (text data), label (classification of the event), and image (associated visual data). The label column indicates whether the post refers to flood, fire, or non-damage events. This multimodal dataset is essential for training models that can handle both textual and visual data to predict disaster-related events.

4.2. Data Collection using Web Scraping

Given the limitations of the Twitter API, Selenium and BeautifulSoup were employed to scrape tweets directly from Twitter’s web interface. This method allows us to collect the necessary tweets, user metadata, and other related information for further processing.

Steps Involved:

1. Login Management: Selenium automates the login process and manages cookies to maintain a valid session.

2. Automated Scrolling: Selenium simulates user

3. scrolling to load older tweets dynamically.

4. Content Extraction: BeautifulSoup is used to extract tweet content, metadata (such as timestamp, user location), and associated images.

5. Proxy and Anti-Bot Mechanisms: Techniques like proxy rotation and user-agent spoofing were implemented to mimic human browsing behavior and bypass Twitter's anti-bot detection.

Once scraped, the data (tweets, metadata, and images) is streamed into Apache Kafka for further processing.

```python
from selenium import webdriver
from bs4 import BeautifulSoup

# Initialize Selenium WebDriver
driver = webdriver.Chrome(executable_path='/path/to/chromedriver')
driver.get('https://twitter.com/search?q=%23disaster')

# Scroll and load tweets
for i in range(10):
    driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
    soup = BeautifulSoup(driver.page_source, 'html.parser')
    tweets = soup.find_all('div', {'data-testid': 'tweet'})
    for tweet in tweets:
        # Extract tweet content, metadata, etc.
        print(tweet.text)
```

Code for Webscraping

4.3. Real-Time Data Ingestion with Apache Kafka

The scraped data is immediately streamed into Apache Kafka for real-time ingestion. Kafka acts as a message broker that handles high-throughput streams of tweets, partitioning the data into different topics based on keywords or geographic locations.

1. Kafka Producer API streams the tweets to respective topics (e.g., “#Flood”, “#Fire”).
2. Partitioning: Tweets are partitioned by disaster types (e.g., “earthquake”, “flood”) to allow for parallel processing.

This partitioning ensures scalable, real-time disaster monitoring.

4.4. Real-Time Data Processing with Apache Spark

Once the tweets are ingested, Apache Spark Streaming processes the data in micro-batches. This real-time processing pipeline performs the following tasks:

```
from pyspark.sql import SparkSession

spark = SparkSession.builder.appName('DisasterTweetProcessing').getOrCreate()
df = spark.readStream.format("kafka").option("kafka.bootstrap.servers",
"localhost:9092").option("subscribe", "disaster_tweets").load()

# Perform sentiment analysis and location extraction
processed_tweets = df.withColumn('sentiment',
analyze_sentiment(df.value)).withColumn('location',
extract_location(df.value))
```

Code for sparksession

1. Sentiment Analysis: NLP techniques are used to assess the sentiment of tweets related to disaster events.
2. Location Extraction: Extracts disaster locations and severity from the tweet content using geospatial analysis and named entity recognition (NER).

By leveraging Spark’s distributed processing capabilities, the system can handle a massive influx of tweets, even during high-traffic periods, ensuring insights are delivered in real-time.

4.5. Multimodal Classification Model

The core of the system is a multimodal classification model designed to process both text and image data. For text classification, the system leverages BERT and XLNet, two powerful natural language processing models that have been fine-tuned on disaster-specific datasets. These models are able to capture the context and semantic meaning from the tweets, making them particularly effective for identifying and classifying disaster-related text content. In parallel, EfficientNet-B3 is used for image classification. This model has been trained on a curated dataset of disaster images, enabling it to accurately categorize visual data into classes such as flood, fire, and non-damage.

To combine the strengths of both text and image data, the system utilizes a model fusion approach. The embeddings produced by the BERT/XLNet models for text and the EfficientNet-B3 model for images are concatenated in a fusion layer. This unified representation of the tweet, which now contains both textual and visual information, is then passed through fully connected layers. The final output is a classification of the tweet, indicating whether it pertains to a flood, fire, or non-disaster event. By fusing these two modalities, the system is able to improve accuracy and leverage the complementary strengths of text and image data.

4.6. Training Procedure

In the training procedure, we utilize CrossEntropyLoss as the loss function, which is well-suited for multi-class classification tasks. This ensures that the model is optimized to correctly classify tweets into disaster categories such as flood, fire, or non-damage. For optimization, we use the AdamW optimizer with a learning rate of 1e-4. AdamW is known for its efficiency in handling large-scale datasets and includes built-in weight decay regularization, which helps prevent overfitting during training.

The model is trained over 20 epochs, with early stopping in place based on the validation accuracy. This ensures that training is halted when no significant improvement is observed in validation performance, reducing overfitting and unnecessary training time. During each epoch, the model is switched to training mode, and the loss is computed using both text and image inputs. The optimizer updates the model weights based on the calculated gradients. For each batch, the loss is computed, backpropagated, and accumulated, allowing us to track the running loss throughout the training process. The loss for each epoch is printed at the end to monitor the model's learning progress

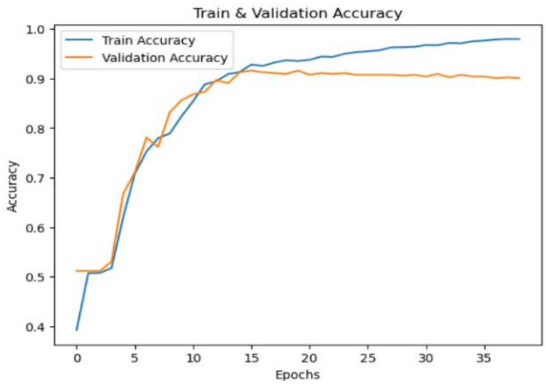

Fig 3. Train & Validation Accuracy Over Epochs

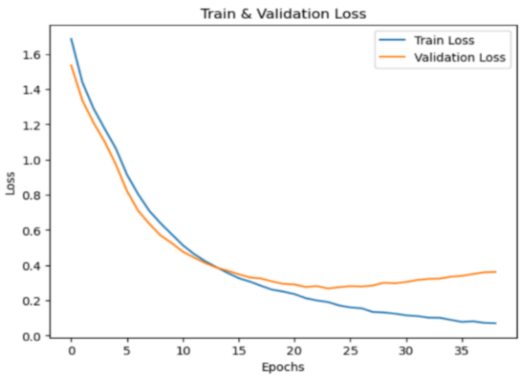

Fig 4. Train & Validation Accuracy Over Epochs

4.7. Evaluation Metrics

The evaluation of the model's performance is carried out using several key metrics. Accuracy provides a basic measure of overall correctness by determining the proportion of correct predictions. To further assess the model's effectiveness in distinguishing between different disaster categories, Precision, Recall, and the F1-Score are used. These metrics provide deeper insights, especially in handling imbalanced data by focusing on the balance between correctly identified disasters and the false positives or negatives.

Additionally, the Confusion Matrix is used to offer a more detailed view of the model's classification performance. It visualizes how well the model is distinguishing between the three classes—flood, fire, and non-damage—by showing the counts of true positives, false positives, true negatives, and false negatives for each class.

Moreover, throughout the training process, we track training loss, validation loss, training accuracy, and validation accuracy. These metrics, plotted over the epochs, help in analyzing the model's convergence behavior, indicating whether the model is improving, overfitting, or underfitting. By monitoring these metrics, we can adjust the model training to optimize its performance for disaster tweet classification.

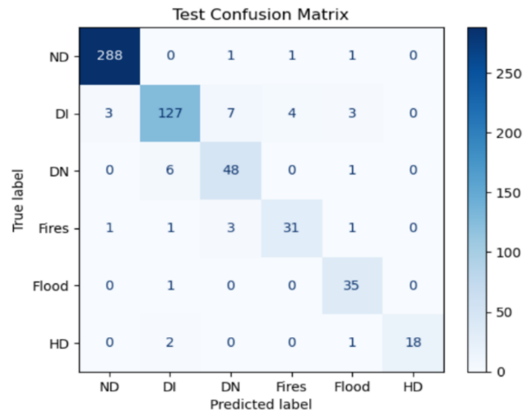

Fig 5. Confusion Matrix

This implementation outlines the full pipeline of our disaster tweet processing system, from web scraping and real-time streaming with Kafka, to multimodal classification using BERT and EfficientNet, tweet filtering with Gemini LLM, and data storage with MongoDB and Elasticsearch. The system delivers real-time disaster insights through a combination of advanced data processing techniques, machine learning models, and visualization tools.

V. RESULTS AND DISCUSSION

The results of the comparative analysis between the EfficientNet-BERT and ResNet-BERT models demonstrate distinct performance characteristics in classifying disaster-related tweets. Both models achieved high accuracy rates, but EfficientNet-BERT outperformed ResNet-BERT across all key metrics, indicating its superior ability to differentiate between relevant and irrelevant tweets in the context of disaster management.

Metrics	EfficientNet-BERT	ResNet-BERT
Accuracy	0.9366	0.9250
Precision	0.9385	0.9200
Recall	0.9366	0.9180
F1 Score	0.9368	0.9190

Table 2. Summary of performance metrics

5.1. Model Performance

The EfficientNet-BERT model achieved an impressive accuracy of **93.66%**, indicating that it correctly classified a high percentage of tweets across the flood, fire, and non-damage categories. This suggests that the model has learned to differentiate effectively between relevant and irrelevant tweets in the context of disaster management. In contrast, the ResNet-BERT model performed well with an accuracy of **92.50%**, but demonstrated slightly lower scores across all metrics compared to

EfficientNet-BERT. This indicates that, while ResNet-BERT is a strong model, it may not be as finely tuned for this specific classification task.

5.2. Precision and Recall

EfficientNet-BERT's precision of **0.9385** signifies a robust ability to minimize false positives, meaning that when it predicts a tweet as relevant to disasters, it is correct **93.85%** of the time. This is crucial in disaster response contexts, where false alarms can lead to unnecessary resource allocation. The recall score of **0.9366** highlights that the model effectively captures the majority of actual disaster-related tweets. This balance between precision and recall is vital for ensuring that significant tweets are not missed, which can be critical during emergencies.

5.3. F1 Score

The F1 Score of **0.9368** for EfficientNet-BERT reflects a harmonious trade-off between precision and recall. It indicates that the model is robust in classifying tweets without being overly sensitive to either false positives or false negatives. On the other hand, the F1 Score of **0.9190** for ResNet-BERT, while commendable, suggests that there may be a slight trade-off in performance. This potentially makes EfficientNet-BERT the more reliable option in scenarios where both precision and recall are equally important.

VI. CONCLUSION AND FUTURE WORK

This research paper presents the development of an AI-driven disaster management system that harnesses real-time social media data to enhance disaster response. By integrating data from platforms like Twitter, Instagram, and Facebook, the system collects both textual and image-based information critical for situational awareness during disasters. The system employs advanced machine learning models, such as BERT and XLNet for text analysis, and EfficientNet and ResNet50 for image classification, to accurately categorize and assess the severity of disaster events. This multimodal approach significantly improves the precision of disaster detection, providing actionable insights that help responders prioritize and allocate resources effectively.

The system’s real-time data ingestion and processing pipeline, built using Apache Kafka and Apache Spark, ensures scalability and efficiency even during high-traffic disaster periods. By fusing textual and visual data, the system achieves high classification accuracy, ensuring that relevant disaster information is quickly identified. The architecture’s scalability allows for processing large volumes of social media data without bottlenecks, making it a robust solution for managing disaster-related data in real time. Ultimately, this system demonstrates the potential of social media analytics in improving disaster response, helping save lives by providing timely and actionable insights.

Looking forward, the system will be expanded into a web portal aimed at providing disaster response agencies in India with a powerful tool for analysing and categorizing disaster-related data from significant repositories, particularly social media. This portal will automate the collection, processing, and analysis of data, delivering real-time actionable insights that allow agencies to respond more quickly and efficiently. By reducing response times, the portal will help agencies save valuable time, money, and resources, while also enhancing the accuracy and relevance of the data used for decision-making. This future development will streamline the process of disaster monitoring and management, offering a centralized platform where agencies can access critical information, ultimately improving disaster preparedness and response.

References

[1] Wiegmann, M., Kersten, J., Senaratne, H., Potthast, M., Klan, F., and Stein, B.: *"Opportunities and risks of disaster data from social media: a systematic review of incident information,"* Nat. Hazards Earth Syst. Sci., 21, 1431–1444, https://doi.org/10.5194/nhess-21-1431-2021, 2021.

[2] A. Pal, J. Wang, Y. Wu, K. Kant, Z. Liu, and K. Sato, *"Social Media Driven Big Data Analysis for Disaster Situation Awareness: A Tutorial,"* IEEE Transactions on Big Data, vol. 9, no. 1, pp. 1-21, Feb. 2023, doi: 10.1109/TBDATA.2022.3158431.

[3] H. Hao and Y. Wang, "Leveraging multimodal social media data for rapid disaster damage assessment," *International Journal of Disaster Risk Reduction*, vol. 51, 2020, Art. no. 101760. https://doi.org/10.1016/j.ijdrr.2020.101760

[4] Jaebeom You, Kisung Lee, Hyuk-Yoon Kwon, "*DeepScraper: A complete and efficient tweet scraping method using authenticated multiprocessing*," Data & Knowledge Engineering, Volume 149, 2024, 102260, ISSN 0169-023X, https://doi.org/10.1016/j.datak.2023.102260.

[5] Domala, Jayashree & Dogra, Manmohan & Masrani, Vinit & Fernandes, Dwayne & D'souza, Kevin & Fernandes, Delicia & Carvalho, Tejal. (2020). *"Automated Identification of Disaster News for Crisis Management using Machine Learning and Natural Language Processing."* 503-508. 10.1109/ICESC48915.2020.9156031.

[6] Mishra, Sameer & Bisen, Atharva & Mundhada, Soham & Singh, Utkarsh & Bongirwar, Vrushali. (2022). *"Disaster Information Verification and Validation Application Using Machine Learning. International Journal of Next-Generation Computing".* 10.47164/ijngc.v13i5.907.

[7] Ningsih, A & Id Hadiana, Asep. (2021). Disaster Tweets Classification in Disaster Response using Bidirectional Encoder Representations from Transformer (BERT). IOP Conference Series: Materials Science and Engineering. 1115. 012032. 10.1088/1757-899X/1115/1/012032.

[8] Kusumasari, B., Prabowo, N.P.A. *"Scraping social media data for disaster communication: how the pattern of Twitter users affects disasters in Asia and the Pacific."Nat Hazards* **103**, (2020). https://doi.org/10.1007/s11069-020-04136-z

[9] M. Laya, M. K. Delimayanti, A. Mardiyono, F. Setianingrum, A. Mahmudah and D. Anggraini, "Classification of Natural Disaster on Online News Data Using Machine Learning," *2021 5th International Conference on Electrical, Telecommunication and Computer Engineering (ELTICOM)*, Medan, Indonesia, 2021, pp. 42-46, doi: 10.1109/ELTICOM53303.2021.9590125.

[10] A., Aswathy., Rekha, Prabha., Lakshmi, Gopal., Divya, Pullarkatt., Maneesha, Vinodini, Ramesh. (2022). *"An efficient twitter data collection and analytics framework for effective disaster management."* doi: 0.1109/DELCON54057.2022.9753627

[11] Meghna, Jayan., Lija, Jacob. (2023). 1. "*Categorizing Disaster Tweets Using Learning Based Models for Emergency Crisis Management."* doi: 10.1109/ICACCS57279.2023.10113105

[12] P., Gnanasekaran. (2023). 1. *"Juncture of Text Preprocessing Techniques & Extracting Sentiment Analyzing of Micro-Blog Based on Machine Learning Algorithms."* doi: 10.1109/icses60034.2023.10465450

[13] Parth, Mahajan., Pranshu, Raghuwanshi., Hardik, Setia., Princy, Randhawa. (2024). 1. *"A Multi-Model Approach for Disaster-Related Tweets."* doi:10.57159/gadl.jcmm.3.2.240125

[14] Pardeep, Singh., Monika, Singh., P., Das., Satish, Chand. (2024). 2. *"Leveraging domain-adapted transformer based models for disaster-specific tweet classification."* doi: 10.3233/jifs-219419

[15] Wei, Chen., Janet, Fang. (2024). 1. *"Optimizing AI-Driven Disaster Management through LLMs."* doi: 10.20944/preprints202407.1446.v1

[16] Frankie, Frankie., Yeremia, Alfa, Susetyo. (2023). 2. *"Implementation of text indexing system in web-based document search application using mongodb."* doi: 10.52436/1.jutif.2023.4.5.959

[17] Siva, Krishna, Dasari., G., Srinivas., P., Jayarama, Reddy. (2024). 2. *"A deep parallel hybrid fusion model for disaster tweet classification on Twitter data."* Decision Analytics Journal, doi: 10.1016/j.dajour.2024.100453

2. Plagiarism Report

Submission date: 11-Nov-2024 02:59PM (UTC+0530)
Submission ID: 2515718523
File name: AI-Powered_Multimodal_Disaster_Response.docx (797.99K)
Word count: 5087
Character count: 31667

AI-Powered Multimodal Disaster Response Enhancement Using Social Media Streams

Sai Thikekar, Arya Banavali, Yash Chhaproo, Aradhya Ingle, Dr. Rohini Temkar

Department of Computer Engineering, Vivekanand Education Society's Institute Of Technology, Mumbai, India

***Abstract*— Social media serves as a valuable source of real-time disaster data, providing timely updates that are critical for effective disaster monitoring and response. It plays an essential role in raising awareness, enhancing preparedness, and facilitating prompt action during emergencies. While much of the current research focuses on disaster prediction, real-time updates, which are key to saving lives and coordinating response efforts, are often overlooked. Our project addresses this gap by focusing on the collection and analysis of real-time data from various social media platforms, including disaster-related images and textual data like tweets.We use multiple APIs to gather real-time data from platforms such as Instagram, Facebook, and Twitter. To analyze visual data, we employ state-of-the-art multimodal models like ResNet50 and EfficientNet for image classification. For the textual data, advanced natural language processing models like BERT and XLNet are used for sentiment analysis, event detection, and severity assessment. By integrating and fusing the image and text data, our system is able to identify the type of disaster, estimate its severity, and provide actionable insights.**

that combining various data types significantly improves the spatial accuracy of event detection, such as urban flooding, achieving up to 100% precision in specific instances ("A Spacel Information Extraction Method based on Multimodal Social Media Data: A Case Study on Urban Inundation," 2023). Moreover, Chatterjee [3] propose an innovative approach for monitoring mental health through social media posts, using sentiment analysis and multimodal features to construct a mental health index, achieving 89% accuracy with Support Vector Machines (SVM). This real-time analysis has profound implications for disaster response, as psychological well-being is a critical aspect of managing the human impact of disasters.

Disaster response can also be enhanced by employing advanced natural language processing models like Late

---

future risks [1].

In recent years, social media platforms, such as Twitter, Instagram, and Facebook, have played an increasingly important role in disaster management by providing real-time data that enhances situational awareness and facilitates rapid emergency response. The integration of social media analytics into disaster management frameworks has revolutionized how information is collected, processed, and acted upon during crises. Social media's ability to provide up-to-date, location-specific data enables authorities and responders to monitor dynamic situations and coordinate more efficient responses [2].

collection component of disaster response systems.

To complement social media data collection, the aggregation of disaster-related news is essential for a comprehensive crisis management system. Domala [5] proposed a system that automates the collection of disaster news using web scraping techniques and classifies the gathered data through machine learning models such as Support Vector Machines (SVM) and Logistic Regression. This system also incorporates Natural Language Processing (NLP) to preprocess the news data and geoparses relevant disaster locations. With an evaluation based on precision, recall, F1 score, and a confusion matrix, the

---

tasks. This ensures that the model is optimized to correctly classify tweets into disaster categories such as flood, fire, or non-damage. For optimization, we use the AdamW optimizer with a learning rate of 1e-4. AdamW is known for its efficiency in handling large-scale datasets and includes built-in weight decay regularization, which helps prevent overfitting during training.
The model is trained over 20 epochs, with early stopping in place based on the validation accuracy. This ensures that

deeper insights, especially in handling imbalanced data by focusing on the balance between correctly identified disasters and the false positives or negatives.

Additionally, the Confusion Matrix is used to offer a more detailed view of the model's classification performance. It visualizes how well the model is distinguishing between the three classes—flood, fire, and non-damage—by showing the counts of true positives, false positives, true negatives, and false negatives for each class.

AI-Powered Multimodal Disaster Response

ORIGINALITY REPORT

1%	2%	1%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to University of Arizona Global Campus (Canvas LTI 1.1) Student Paper	1%
2	"Technology Innovation for Sustainable Development of Healthcare and Disaster Management", Springer Science and Business Media LLC, 2024 Publication	1%

Exclude quotes	On	Exclude matches	< 1%
Exclude bibliography	On		

3. Review Sheet Screenshots

Review 1 sheet

Inhouse/ Industry _Innovation/Research:

Sustainable Goal:

Project Evaluation Sheet 2024 - 25

Class: D17 A/B/C

Group No.: 19

Title of Project: ResQconnect-AI Disaster Management Application

Group Members: Arya Banavali (1), Aaradhya Ingle (24), Sai Thikkekar (64), Yash Chapproo (7)

Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (2)	Applied Engg&Mgmt principles (3)	Life - long learning (3)	Professional Skills (3)	Innovative Approach (3)	Research Paper (5)	Total Marks (50)
5	5	5	3	5	2	2	2	2	2	3	3	3	2	4	48

Comments:

Name & Signature Reviewer1

Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (2)	Applied Engg&Mgmt principles (3)	Life - long learning (3)	Professional Skills (3)	Innovative Approach (3)	Research Paper (5)	Total Marks (50)
05	05	05	03	05	02	02	02	02	02	03	03	03	02	04	48

Comments:

Date: 1st March, 2025

Name & Signature Reviewer 2

Review 2 Sheet

Inhouse/ Industry _Innovation/Research:

Sustainable Goal:

Project Evaluation Sheet 2024 - 25

Class: D17 A/B/C

Group No.: 19

Title of Project: ResQconnect - AI Driven Disaster Management System

Group Members: Arya Banavali (1), Aaradhy Ingle (24), Yash Chhaproo (7), Sai Thikkekar (64)

Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (2)	Applied Engg&Mgmt principles (3)	Life - long learning (3)	Professional Skills (3)	Innovative Approach (3)	Research Paper (5)	Total Marks (50)
04	04	04	03	04	02	02	02	02	02	03	03	03	02	04	44

Comments:

Name & Signature Reviewer1

Engineering Concepts & Knowledge (5)	Interpretation of Problem & Analysis (5)	Design / Prototype (5)	Interpretation of Data & Dataset (3)	Modern Tool Usage (5)	Societal Benefit, Safety Consideration (2)	Environment Friendly (2)	Ethics (2)	Team work (2)	Presentation Skills (2)	Applied Engg&Mgmt principles (3)	Life - long learning (3)	Professional Skills (3)	Innovative Approach (3)	Research Paper (5)	Total Marks (50)
04	04	04	03	04	02	02	02	02	02	03	02	03	02	04	43

Comments: Good Work. Can include tracking feature in In Progress option.

Date: 1st April,2025

Name & Signature Reviewer 2

II. Competition certificates:


Vishwaniketan's Institute of Management Entrepreneurship and Engineering Technology (ViMEET)

(Affiliated to University of Mumbai/Approved by DTE/AICTE/NAAC at A+ Grade in 1st Cycle)

VISHWANIKETAN

RKDEMY

0x.Day

Aptech LEARNING

MARKSCAPE

Hack To Crack 2.0

CERTIFICATE

Of participation

Awarded to

YASH CHHAPROO

of VIVEKANAND EDUCATION SOCIETY INSTITUTE OF TECHNOLOGY

for participating in

National Level Hackathon Hack To Crack 2.0

20-21 March 2025

Dr. Ankush Pawar
CONVENER

Dr. B R Patil
PRINCIPAL

HTTPS://HACKTOCRACK.IN/VERIFY-CERTIFICATE

H2C/2025/205


Vishwaniketan's Institute of Management Entrepreneurship and Engineering Technology (ViMEET)

(Affiliated to University of Mumbai/Approved by DTE/AICTE/NAAC at A+ Grade in 1st Cycle)

VISHWANIKETAN

RKDEMY

0x.Day

Aptech LEARNING

Hack To Crack 2.0

CERTIFICATE

Of participation

Awarded to

SAI THIKEKAR

of VIVEKANAND EDUCATION SOCIETY INSTITUTE OF TECHNOLOGY

for participating in

National Level Hackathon Hack To Crack 2.0

20-21 March 2025

Dr. Ankush Pawar
CONVENER

Dr. B R Patil
PRINCIPAL

HTTPS://HACKTOCRACK.IN/VERIFY-CERTIFICATE

H2C/2025/203