Emergency AI System for Real-Time Disaster Message Analysis and Classification

Authors:
Srinivasan S, Usha Devi M
BCA DevOps & Automation, Rathinam College of Arts and Science, India

srinivasan.sriraman2346@gmail.com,
usha.devi145@gmail.com

ABSTRACT

This paper presents an Emergency AI system designed to analyze and classify emergency-related messages using natural language processing and machine learning techniques. Traditional emergency communication systems rely primarily on phone-based interactions, which may lead to delays, miscommunication, and inefficiency during critical situations. In large-scale emergencies, handling multiple incoming calls becomes challenging, reducing the effectiveness of response systems.The proposed system provides a web-based platform where users can submit emergency-related messages in text format. These messages are processed using a multi-stage pipeline consisting of text preprocessing, feature extraction using TF-IDF, emergency detection, category classification, and severity analysis. The system converts unstructured text into structured information that can be easily interpreted by authorities.The processed results are stored in a database and displayed on a centralized dashboard. This enables real-time monitoring, efficient prioritization of incidents, and faster response. The system demonstrates how artificial intelligence can enhance emergency management by improving communication efficiency and decision-making capabilities.

INTRODUCTION

Emergency response systems are critical for ensuring public safety during disasters and crisis situations. Traditional systems rely heavily on voice-based communication such as phone calls, which may not always be reliable. During peak situations, communication networks may become overloaded, leading to delays in response. Additionally, information provided through calls may be incomplete or unclear, making it difficult for authorities to assess the situation accurately.With the advancement of artificial intelligence and web technologies, there is an opportunity to improve emergency communication systems. The Emergency AI system is designed to provide an intelligent solution for analyzing emergency messages in real time. Instead of

relying solely on calls, users can submit messages through a web interface, which are automatically processed by the system.The system uses machine learning algorithms to detect whether a message represents an emergency, classify it into categories such as fire, medical, or accident, and determine its severity level. This structured output enables authorities to prioritize incidents and respond more effectively. The system also reduces dependency on manual analysis, ensuring consistent and accurate interpretation of information.

EXISTING SYSTEM

The existing emergency response system primarily depends on voice-based communicationthrough emergency helpline numbers such as police, ambulance, and fire services. In this system, users reportincidents by making phone calls, and operators manually analyze the information provided.These systemstypically rely on human operators to understand the situation, categorize the incident, and forward it to theappropriate department. While this method has been widely used and is effective in many cases, it has severallimitations when dealing with large-scale emergencies or high volumes of incoming calls.Traditional systemsuse basic data recording methods, and most of the information remains unstructured. There is no automated mechanism to analyze incoming messages or prioritize incidents based on severity. As a result, criticalsituations may not always receive immediate attention.Additionally, during peak times or disaster situations,emergency call centers often experience heavy traffic, leading to call delays or dropped calls. This reduces the efficiency of emergency response systems and increases the risk of delayed action.

DRAWBACKS FOR EXISTING SYSTEM

The existing emergency system has several limitations that affect its performance and efficiency.One of the major drawbacks is call congestion, especially during disasters or peak times. When many users tryto contact emergency services simultaneously, the system becomes overloaded, leading to delays in communication.Another limitation is the lack of structured data processing. Information provided through calls is not stored in a standardized format, making it difficult to analyze and prioritize incidents.The system also depends heavily on human interpretation, which can lead to errors in understanding the situation. Miscommunication or incomplete information may result in incorrect categorization of incidents.Additionally,there is no automated prioritization mechanism to identify high-severity cases. All incidents are treatedsimilarly, which can delay response to critical situations.The absence of a centralized monitoring dashboardfurther limits the ability

of authorities to track and manage incidents effectively.

PROPOSED SYSTEM

The proposed system introduces an AI-based approach to emergency management by enabling users to submit incident information in text format through a web application. The system processes thesemessages using Natural Language Processing and Machine Learning techniques to automatically analyze andclassify the information.The system performs multiple stages of processing. First, it determines whether themessage represents an emergency. If the message is classified as an emergency, it is further analyzed to identifythe category of the incident, such as fire, medical emergency, flood, or gas leakage. In addition to classification,the system assigns a severity level based on the urgency and context of the message. This helps in prioritizingincidents and ensuring that critical cases receive immediate attention.All processed data is stored in a database and displayed on a government monitoring dashboard. The dashboard provides a clear view of all incidents, including their category, severity, and timestamp. Authorities can use this information to make informed decisions and allocate resources efficiently.The system is designed to be scalable and can be deployed on cloud platforms such as AWS. It supports real-time processing and can handle multiple user inputs simultaneously.

ADVANTAGES OF PROPOSED SYSTEM

It provides automated emergency detection, reducing the need for manual analysis.

- It enables faster prioritization of incidents based on severity.
- It supports real-time monitoring through a centralized dashboard.
- It reduces call center overload by allowing text-based reporting.
- It improves accuracy and consistency in classification using machine learning models.
- It is scalable and cost-effective, making it suitable for large-scale deployment.
- It enhances decision-making capabilities for authorities.

FEASIBILITY STUDY

Technical Feasibility

The technical feasibility of the Emergency AI system is high due to the use of well-establishedand widely supported technologies such as Python, Django framework, and machine learning libraries likeScikit-learn, Pandas, and NumPy. These technologies are open-source, regularly updated, and have

strongcommunity support, making development and maintenance efficient.The system utilizes Natural LanguageProcessing techniques to analyze text-based inputs, which can be effectively implemented using existinglibraries without the need for complex infrastructure. Additionally, the system can be deployed on cloudplatforms such as AWS EC2, ensuring scalability and accessibility. The hardware requirements are minimal andcan be handled by standard computing systems, making the implementation practical even with limitedresources.Overall, the availability of tools, ease of integration, and compatibility across platforms make thesystem technically feasible.

Behavioral Feasibility

The behavioral feasibility of the system is favorable as it is designed with a user-friendlyinterface that requires minimal technical knowledge. Users can easily submit emergency messages through asimple text input field without needing any specialized training.For government authorities, the dashboardprovides a clear and structured view of emergency incidents, including category, severity, and timestamp, whichsimplifies monitoring and decision-making. The system reduces the complexity of analyzing unstructured dataand presents information in an understandable format.As people are increasingly familiar with web applicationsand digital platforms, the adoption of this system is expected to be smooth. The system also builds trust byproviding quick and consistent responses, encouraging users to rely on it during emergency situations.

Economic Feasibility

The system is economically feasible as it is developed using open-source technologies, whicheliminates licensing costs. Tools such as Python, Django, and machine learning libraries are freely available,reducing the overall development expense.Deployment can be carried out using cloud services like AWS EC2,which offers cost-effective solutions and even free-tier options for initial stages. The maintenance cost is also low, as updates and improvements can be implemented without requiring expensive infrastructure.By improvingemergency response efficiency and reducing delays, the system can indirectly contribute to cost savings inresource management and operational activities. Therefore, the benefits of the system outweigh the development and deployment costs.

Implementation:

The system is implemented using Python and Django framework. Django is used to develop the backend and manage the web application, while Django REST Framework is used to build API endpoints for

handling requests.The NLP pipeline is developed using Scikit-learn, which provides tools for text processing and machine learning. Models are trained on emergency-related datasets to ensure accurate classification. The system uses a SQLite database to store processed results.The application provides a user-friendly interface for input and a dashboard for monitoring results. The system supports real-time interaction and can be deployed on cloud platforms such as AWS EC2. The modular design ensures scalability, maintainability, and ease of integration with additional services.

METHODOLOGIES:

The system follows a structured multi-stage approach for analyzing emergency messages:

A. **Text Preprocessing**

The input message is cleaned by removing special characters, converting text to lowercase, and eliminating unnecessary noise. This improves the quality of data for further processing.

B. **Feature Extraction**

TF-IDF (Term Frequency–Inverse Document Frequency) is used to convert textual data into numerical vectors. This enables machine learning models to process the input effectively.

C. **Emergency Detection**

A classification model is used to determine whether the message represents an emergency. This step filters non-emergency messages and focuses only on relevant inputs.

D. **Category Classification**

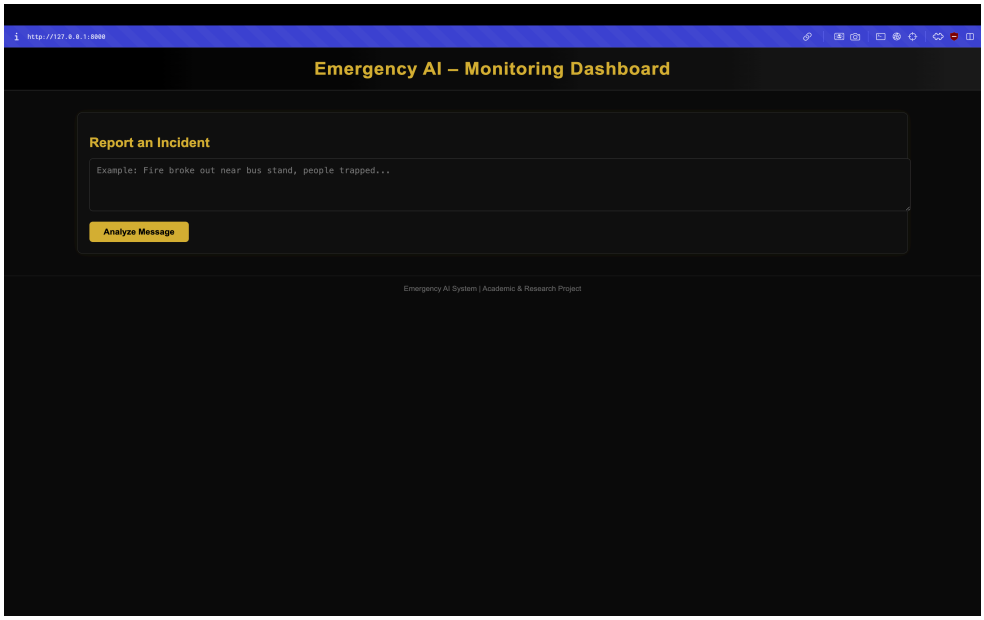
Messages identified as emergencies are classified into categories such as fire, medical, flood, or accident. This helps authorities understand the nature of the incident.

E. **Severity Analysis**

Severity levels are assigned using rule-basedlogic based on keywords and context. Higher severity indicates more critical situations that require immediate attention.

Results and Discussion:

The system successfully processes user messages and generates structured outputs including emergency status, category, severity, and confidence score. The results are displayed on a dashboard, providing a clear view of all incidents.The use of machine learning significantly improves efficiency by automating message analysis. Authorities can quickly identify high-priority cases and respond accordingly. The system reduces manual workload and ensures consistency in decision-making.However, the accuracy of the system depends on the quality of training data and variations in user input. Messages with ambiguous or unclear content may affect classification accuracy. Continuous improvement in training data and model optimization can enhance system performance.



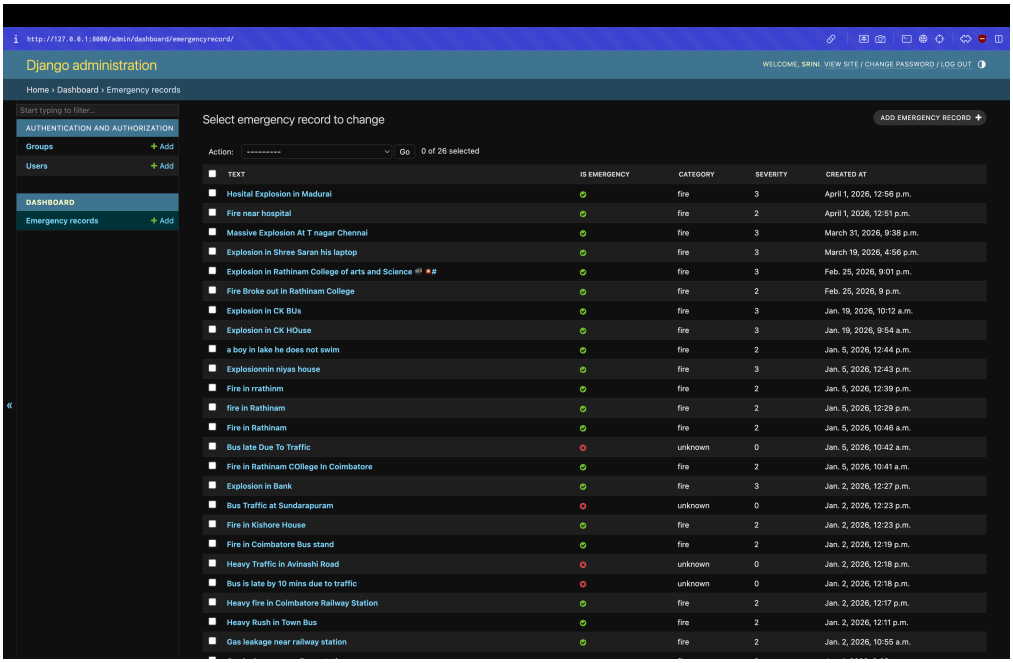
User Interface

```
srini@Srinivasans-MacBook-Air ~ % curl -X POST http://127.0.0.1:8000/api/analyze/ \
-H "Content-Type: application/json" \
-d '{"message":"Fire near hospital"}'
{"text":"Fire near hospital","is_emergency":true,"category":"fire","severity":2,"confidence":0.53}%
srini@Srinivasans-MacBook-Air ~ %
```

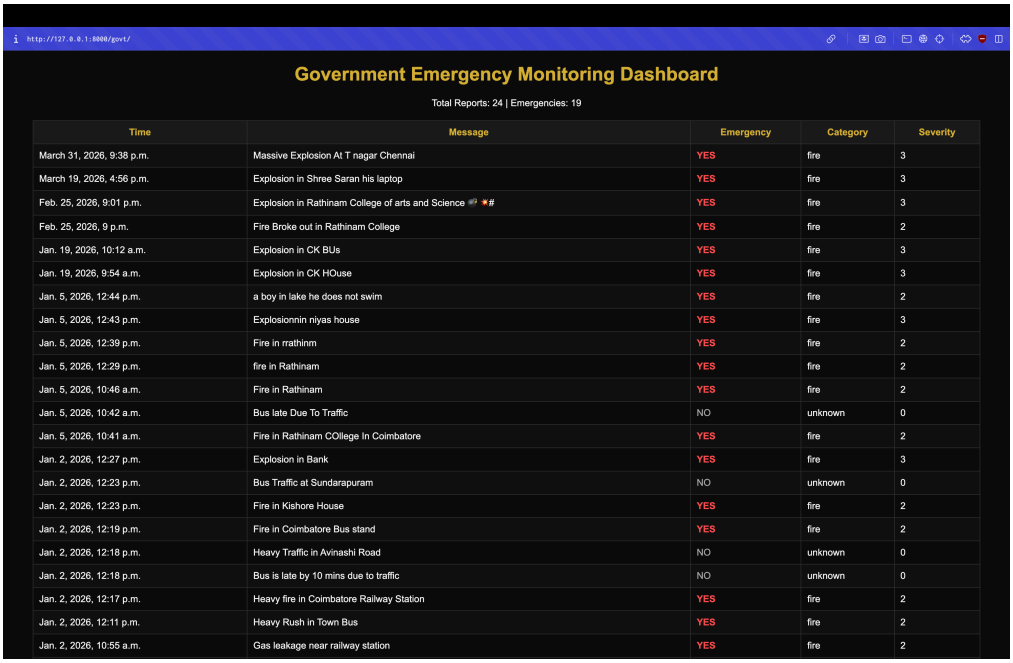
API Module

```
User Message
     ↓
Text Preprocessing
     ↓
TF-IDF Vectorization
     ↓
Emergency Detection
     ↓
Category Classification
     ↓
Severity Calculation
```

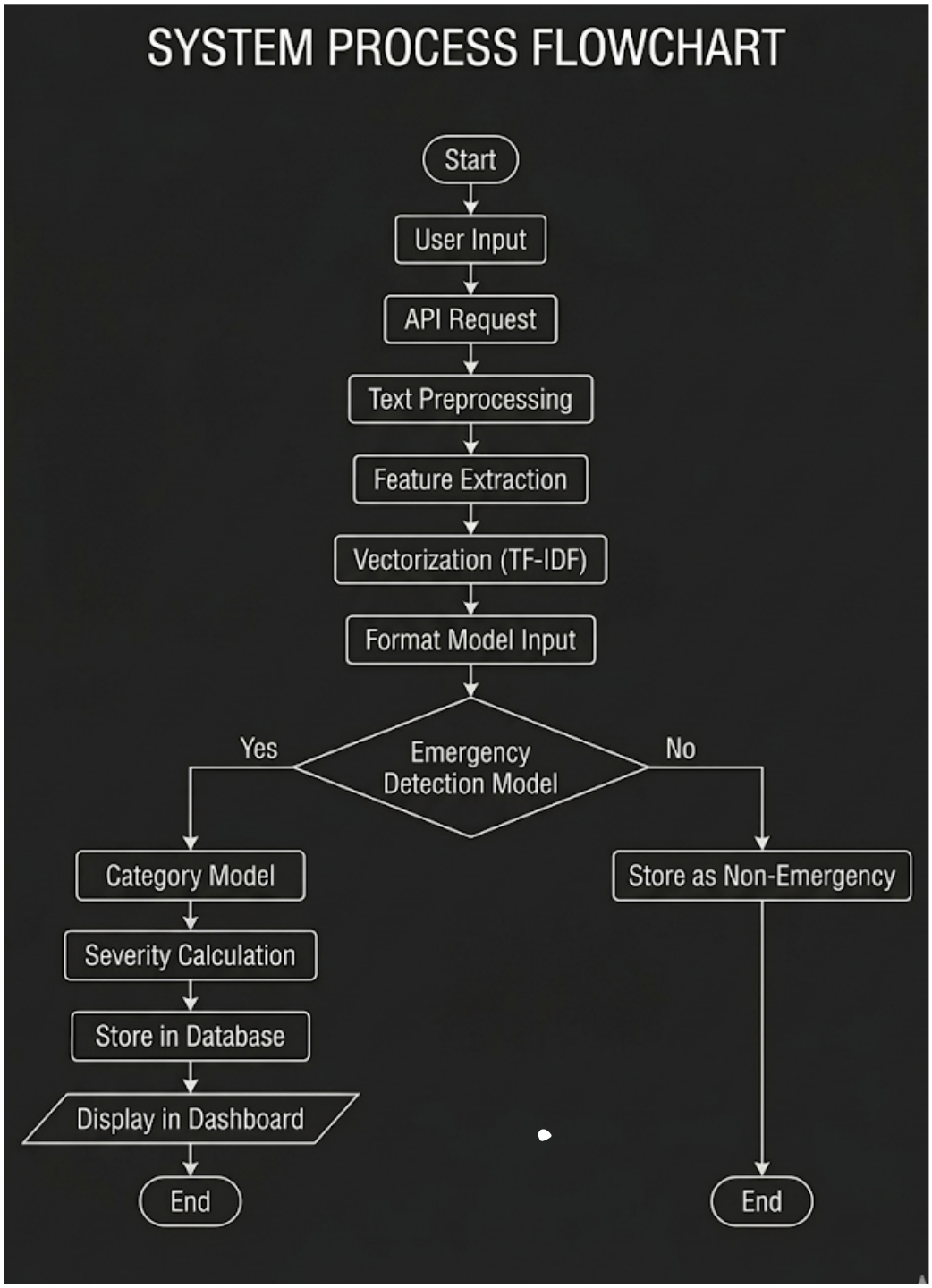
NLP Processing Module



Database Module



Dashboard Module



System Flow Diagram