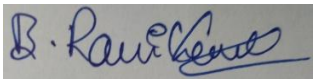
Disaster Tweets Analyzer
Enhanced by NLP and LLMs
(As a part of ACTS Management System)

Functional Requirement Specification Document

05-07-2024

Functional Requirements Document

Name of Document	**Functional Requirement Specification Disaster Tweets Analyzer: Enhanced by NLP and LLMs**
Author	**Mansi, Ajay Singh, Dhananjay Singh, Damini Choudhary, Ritik Verma**
Description of Content	**Functional, Technical and Operational Requirements**
Reference	

Approved by	**Mr. B Reddy Ravi Kumar** 
Date of Approval	**10 July 2024**

Distribution

Copy No.	Distributed to
1	
2	
3	
4	

Document Version Control

Version Number	Version Date	Prepared by	Reviewed by	Particulars/ Description for Change	Remarks

List of Requirements Elicitation Discussions

SN.	Date	Participating Institutes/ Departments
1.		
2.		
3.		
4.		
5.		
6.		
7.		
8.		
9.		
10.		
11.		
12.		
13.		
14.		
15.		
16.		
17.		

Abbreviations

CONTENTS

1. Introduction

1.1. Purpose

The purpose of this document is to provide a detailed description of the requirements for the project "Disaster Tweets Analyzer Enhanced by NLP and LLMs". It aims to illustrate the purpose, scope, and functionality of the system, explaining the system constraints, interfaces, and interactions with other external applications. This document is intended for the customer for functional requirements approval and as a reference for the development team.

1.2. Background

This project aims to leverage Natural Language Processing (NLP) techniques, specifically the BERT model, to analyse tweets related to disasters. By doing so, it will help in the real-time identification of disaster events and sentiments, thereby aiding in disaster management and response.

1.3 Scope

The project will:

- Collect and preprocess tweets related to disaster events.
- Use the BERT model to classify tweets and analyse sentiment.
- Store and manage the processed data for further analysis.

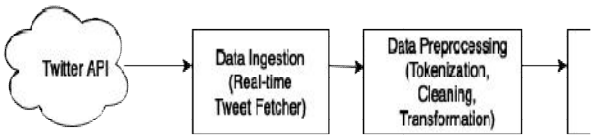


Figure 1: System Architecture

1.4 References

a) BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding by Jacob Devlin et al.

b) Twitter API documentation

1.5 Document Overview

This document provides an overview of the project "Disaster Tweets Analyzer Enhanced by NLP and LLMs" including its interactions, functionality, target audience, and user interface. It describes the functionalities, system constraints, assumptions, dependencies, and requirements.

2. FUNCTIONAL REQUIREMENTS

2.1 Tweet Collection

2.1.1 Description

Collect tweets related to disaster events using specific keywords and hashtags.

2.1.2 Functional Requirements Identified

Functionality 1			
SN.	**Functionality**	**Process**	**Remarks/ Additional Info**
FR 1.1	Tweet Collection	Use Twitter API to collect tweets in real-time	Requires Twitter API keys
FR1.2	Data Preprocessing	Clean and preprocess collected tweets.	Includes text cleaning, removing URLs, fixing abbreviations, and correcting spellings
FR1.3	Vectorization	Convert text data into numerical vectors	Required for initial feature extraction
FR1.4	Model Training	Train deep learning models (Bi-directional LSTM, BERT)	Use BERT model in TensorFlow Keras
FR1.5	Tweet Classification	Classify new tweets as disaster-related or not	Based on the trained BERT model
FR1.	Model Evaluation	Evaluate model performance using appropriate metrics	Precision, recall, F1 score, etc

6			
FR1.7	Data Visualization	Visualize data and model performance metrics	Use libraries like Matplotlib or Seaborn

2.1.3 Fields Validations

This table lists the various fields that will be there in the form for the requirement

SN	Field Name	Field Description	Validations	Remarks
1	Tweet Content	Content of the tweet	Cannot be empty	Mandatory
2	Classification	Disaster category and sentiment	Must be valid category	Mandatory

2.1.4 Pre-requisites, Assumptions and Dependencies

- Twitter API keys must be obtained and configured
- Python libraries for text preprocessing must be installed
- Keywords and hashtags for filtering must be predefined
- Language and geolocation detection algorithms must be implemented
- Sentiment analysis tool must be integrated and configured

2.1.5 Cross Functional Diagram

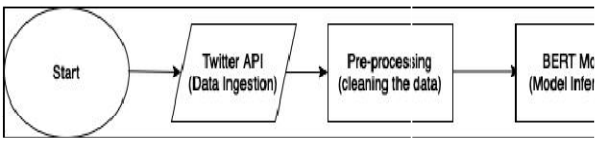


Figure 2: Cross Functional Diagram

1. **Data Ingestion**
   • **Twitter API**: The system connects to the Twitter API to collect real-time tweets. Filters are applied to capture tweets relevant to disasters.

2. **Data Preprocessing**
   • **Cleaning:** Remove unwanted characters, stop words, and perform other cleaning tasks.
   • **Transformation:** Convert text to a format suitable for model input (e.g., converting text to embeddings).

3. **Model Inference**
   • **BERT Model:** Use the BERT model to analyse the pre-processed tweets. This involves feeding the text through the model to get predictions (e.g., classification of tweet content as disaster-related or not).

4. **Post-processing analysis**
   • **Post-processing: A**pply additional logic to the model's output, such as aggregating results or extracting specific information.
   • Disaster **sentiment Analysis**: Determine the sentiment of tweets to gauge public emotion regarding the disaster.

5. **Visualization and Reporting**
   • **Dashboards**: Create dashboards to visualize the data, including real-time updates on disaster-related tweets, sentiment trends, and geographical distributions.
   • **Reporting Tools**: Generate reports to summarize findings and provide insights to stakeholders.