# Crisis Detection from Social Media using Machine Learning

Mehul Pahuja
Indraprastha Institute of Information Technology
mehul22295@iiitd.ac.in

Adya Aggarwal
Indraprastha Institute of Information Technology
adya22043@iiitd.ac.in

Rahul Jha
Indraprastha Institute of Information Technology
rahul22389@iiitd.ac.in

**Project Repository:** Github Link

## Abstract

*In crisis situations, timely information is crucial for situational awareness and effective response. Social media, especially Twitter, offers real-time updates on disaster developments, recovery, and preparedness. This study leverages machine learning to automatically classify crisis-related tweets, enhancing disaster management strategies. We evaluated multiple models, including Logistic Regression, Decision Trees, Random Forest, XGBoost, MLP and CNN. Our findings show that ensemble methods like Random Forest and XGBoost, and MLP offer higher F1 score and scalability, making them suitable for real-time crisis detection. We developed a web-based tool to extract live Tweets, and classify them as disaster or not.*

## 1. Introduction

Social media platforms like Twitter play a crucial role in providing real-time updates during crises. With over 500 million tweets sent daily, Twitter enables users to report incidents and share critical information as events unfold. This makes it a valuable tool for emergency response and disaster management. However, manually processing the vast volume of data is impractical, highlighting the need for automated machine learning approaches.

This project aims to develop a machine learning model to automatically identify and classify crisis-related tweets. By leveraging feature engineering, ensemble learning, and dimensionality reduction, the model seeks to improve crisis detection F1 score and scalability. The ultimate goal is to create a real-time visualization dashboard to support timely decision-making and response during emergencies.

## 2. Literature Survey

Various studies have leveraged social media data, especially Twitter, for real-time disaster detection using machine learning techniques. Key research includes:

### 2.1. Ashktorab [1] *et al*. (2014)

Developed Tweedr, a system leveraging classical machine learning techniques, including Random Forests and Gradient Boosted Trees, to identify and classify disaster-related tweets. By incorporating features such as n-grams, part-of-speech tags, and disaster-specific keywords, their ensemble models demonstrated superior precision and recall, highlighting the importance of domain-specific feature engineering for real-time applications.

### 2.2. Chaudhari & Govilkar [2] (2015)

This paper provides an extensive overview of machine learning techniques applied to sentiment classification, focusing on how these methods process and analyze textual data to identify sentiments like positive, negative, or neutral. The authors explore various traditional and advanced approaches, emphasizing feature selection, classification methods, and challenges in sentiment analysis.

### 2.3. Nguyen [3] *et al*. (2016)

Proposed a CNN-based model for disaster tweet classification, with convolutional and pooling layers to capture local textual features. Their approach achieved a high F1 score, showing the effectiveness of CNNs for processing short, informal text in real-time disaster response scenarios.

## 3. Methodology

Several machine learning models were employed and evaluated for this project. These included:

- **Logistic Regression:** A basic linear model to classify tweets based on the extracted features.

- **Decision Tree:** A tree-based model to capture non-linear relationships between the features.
- **Random Forest:** An ensemble model to improve classification performance by averaging multiple decision trees.
- **XGBoost:** An optimized gradient boosting algorithm for efficient and accurate classification.
- **AdaBoost:** Another boosting technique to focus on harder-to-classify samples.
- **K-Nearest Neighbors (KNN):** A simple instance-based learning algorithm to classify tweets based on distance measures.
- **Support Vector Machine (SVM):** A robust classifier capable of handling high-dimensional feature spaces.
- **Multilayer Perceptron (MLP):** A feedforward neural network with one or more hidden layers, used to capture complex patterns in the data.
- **Convolutional Neural Network (CNN):** A deep learning model designed to extract spatial and local features from tweet text using convolutional operations.

A grid search approach was used to fine-tune the hyperparameters for each model. Additionally, dimensionality reduction was performed using PCA to enhance model efficiency.

Below is the section for the Exploratory Data Analysis (EDA) based on the images and insights provided:

## 4. Exploratory Data Analysis

In this section, we conducted exploratory data analysis (EDA) to gain insights into the distribution and characteristics of tweets within the dataset. The EDA focused on understanding the composition of crisis and non-crisis tweets, as well as the relationships between various features. Key visualizations and findings are detailed below:

### 4.1. Distribution of Crisis and Non-Crisis Tweets

We visualized the distribution of crisis and non-crisis tweets in the dataset and found a relatively balanced number of tweets in both classes, which is crucial for unbiased model training.

### 4.2. Distribution of Tweet Lengths

We explored the distribution of tweet lengths. Most tweets in the dataset are relatively short, with a large concentration of tweets having a length below 200 characters. This observation indicates that most users tend to share concise messages.

### 4.3. Word Cloud for Non-Crisis Tweets

We presented a word cloud visualization of the most frequent words found in non-crisis tweets. The most prominent words include generic and conversational terms, indicating general discussions not necessarily related to crisis events.

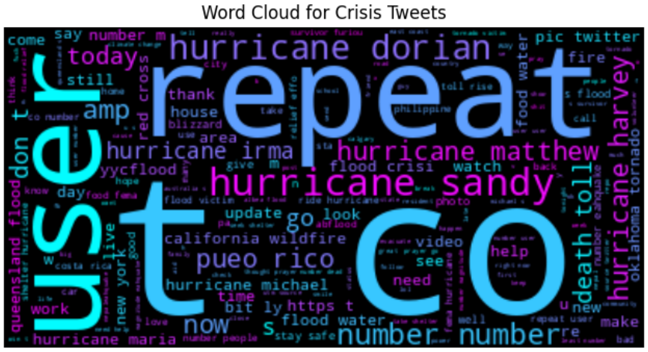

Figure 1. Word Cloud for Non-Crisis Tweets

### 4.4. Sentiment Polarity Distribution

To better understand the sentiment in tweets, we plotted the distribution of sentiment polarity scores. The majority of tweets exhibit neutral sentiment, with fewer tweets exhibiting extreme positive or negative sentiment values.

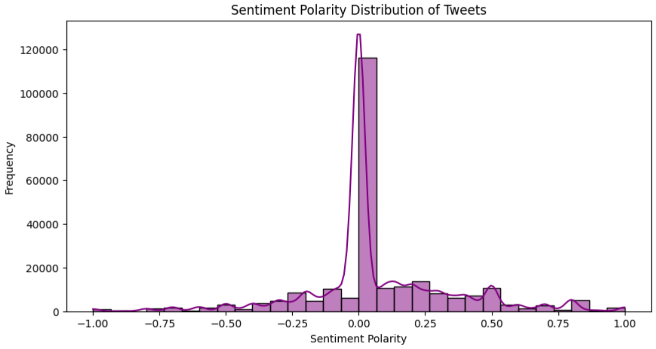

Figure 2. Sentiment Polarity Distribution of Tweets

### 4.5. Sentiment Polarity by Class

We also compared sentiment polarity between crisis and non-crisis tweets. It showed that non-crisis tweets tend to have slightly higher sentiment scores on average, while crisis tweets are generally closer to neutral sentiment.

### 4.6. Common Bigrams Analysis

We analyzed the top 10 most common bigrams (two-word phrases) for crisis and non-crisis tweets. Figures 5 and 5 illustrate that the most frequent bigrams in crisis

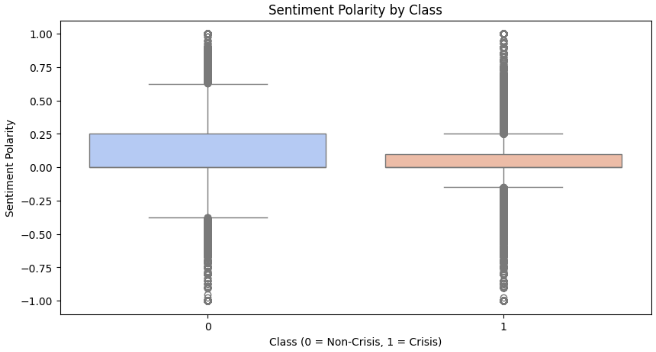

Figure 3. Sentiment Polarity by Class

tweets include specific event-related terms such as "hurricane sandy" and "death toll." In contrast, non-crisis tweets frequently contain general conversational bigrams such as "number number" and "user user."

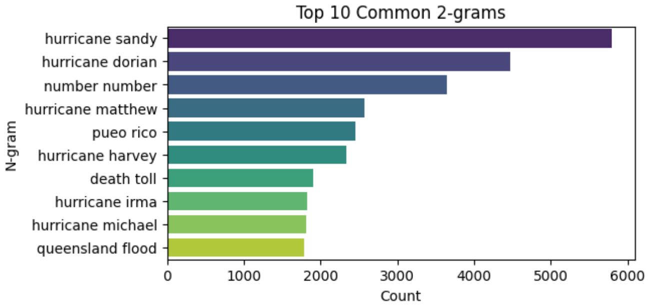

Figure 4. Top 10 Common Bigrams for Crisis Tweets

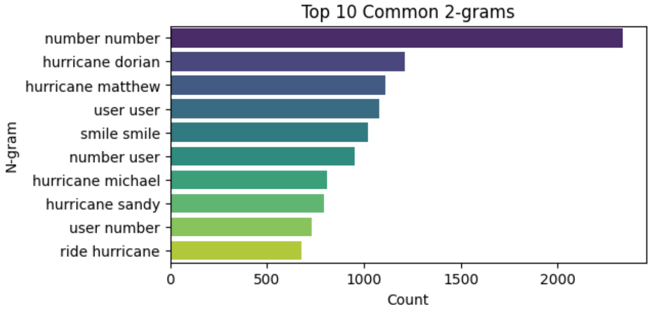

Figure 5. Top 10 Common Bigrams for Non-Crisis Tweets

### 4.7. Average Tweet Length by Class

We examined the average length of tweets for each class. Crisis-related tweets tend to be longer on average compared to non-crisis tweets, which aligns with the nature of detailed reporting during crises.

### 4.8. Correlation Matrix for Numerical Features

We computed and visualized the correlation matrix for key numerical features including tweet length, word count, and sentiment polarity. There is a strong positive correlation between tweet length and word count, while sentiment shows minimal correlation with these features.

## 5. Dataset

The dataset used for this project comprises a total of 247,000 tweets, categorized into crisis and non-crisis classes. Out of these, 118,000 tweets are labeled as non-crisis, while the remaining tweets are categorized as crisis-related. The dataset was created by combining multiple publicly available sources, which are listed below:

- **Twitter Data on Disaster-Related Tweets** (228,005 tweets): This dataset contains a large collection of tweets related to various disaster events. It was obtained from the Omdena platform, which provides data for social good projects. *[Link: https://datasets.omdena.com/dataset/twitter-data-on-disaster-related-tweets]*
- **VStepanenko Disaster Tweets Dataset** (11,380 tweets): This dataset, sourced from Kaggle, includes tweets related to disaster events. It was curated to help train machine learning models for disaster response applications. *[Link: https://www.kaggle.com/datasets/vstepanenko/disaster-tweets/data]*
- **Kaggle NLP Getting Started Competition Dataset** (8,562 tweets): This dataset consists of disaster-related tweets and was originally used for a Kaggle competition focused on natural language processing tasks. *[Link: https://www.kaggle.com/competitions/nlp-getting-started/data]*

## 6. Results

The results were evaluated using F1 score, classification reports, and confusion matrices. Logistic Regression achieved an F1 score of 91.07%, demonstrating its effectiveness as a baseline model. The Decision Tree model achieved an F1 score of 90.87%, indicating its ability to capture non-linear patterns. The Random Forest model provided an improved F1 score of 92.24%, demonstrating its robustness and effectiveness in complex scenarios. The XGBoost model achieved an F1 score of 91.11%, highlighting its capability to handle large feature spaces and its high efficiency.

Support Vector Machine (SVM) outperformed several models, achieving an F1 score of 91.43%. Although the K-Nearest Neighbors (KNN) model had a lower F1 score of 78.83%, it proved effective in specific parameter settings.

When dimensionality reduction was applied using PCA, the results varied. Random Forest, XGBoost, and SVM models experienced slight reductions in F1 score, while KNN and Logistic Regression exhibited moderate performance drops. The highest F1 score achieved using PCA was 90.36% with XGBoost, showing that dimensionality reduction can affect model performance differently depending on the classifier used.

Table **??** shows the classification performance metrics (Precision, Recall, and F1-Score) for both classes (0 and 1), with all values being 0.93, indicating balanced performance across both classes. The F1-Score, Macro Average, and Weighted Average are also reported as 0.93, suggesting a strong and consistent model performance.

The CNN model achieved an F1 score of 89.2% (CNN F1 score: 0.892), demonstrating a strong classification performance, but weaker than some other models.

The MLP Accuracy was MLP Accuracy: 0.9268

The baseline Random Forest model achieved an accuracy of 65.91% without using TF-IDF. This was a simpler model, based on basic features such as tweet length, word count, and sentiment, without considering text content.

The baseline K-Nearest Neighbors (KNN) model with TF-IDF features achieved an F1 score of 78.83%, and with the addition of PCA, it achieved an F1 score of 84.21%.

## 7. Conclusion

This study explored various machine learning models for classifying crisis-related tweets. Ensemble models, especially Random Forest and XGBoost, outperformed simpler models, showing robustness and efficiency for real-time crisis detection. With further tuning and larger datasets, these models can be effectively deployed in practical applications.

Dimensionality reduction (PCA) had varying effects across classifiers, with minimal negative impact on most models. While neural networks like MLPs performed well, they were slightly weaker compared to ensemble methods, and CNNs performed less effectively than others.

The models were integrated into a web-based tool for visualizing disaster-related tweets, enabling real-time crisis monitoring. Overall, Random Forest and XGBoost present promising solutions for scalable crisis detection, while dimensionality reduction and neural networks offer opportunities for further optimization.

| **Model** | **Embedding** | **F1 score (Test)** |
|---|---|---|
| Logistic Regression | TF-IDF | 0.9107 |
| Logistic Regression | TF-IDF + PCA | 0.9015 |
| Decision Tree | TF-IDF | 0.9087 |
| Decision Tree | TF-IDF + PCA | 0.8233 |
| Random Forest | TF-IDF | 0.9224 |
| Random Forest | TF-IDF + PCA | 0.8909 |
| XGBoost | TF-IDF | 0.9111 |
| XGBoost | TF-IDF + PCA | 0.9036 |
| AdaBoost | TF-IDF | 0.8876 |
| AdaBoost | TF-IDF + PCA | 0.8536 |
| K-Nearest Neighbors (KNN) | TF-IDF | 0.7883 |
| K-Nearest Neighbors (KNN) | TF-IDF + PCA | 0.8421 |
| Support Vector Machine (SVM) | TF-IDF | 0.9143 |
| Support Vector Machine (SVM) | TF-IDF + PCA | 0.8099 |
| MLP | TF-IDF | 0.9268 |
| MLP | TF-IDF + PCA | 0.9200 |
| CNN | TF-IDF | 0.8968 |
| CNN | TF-IDF + PCA | 0.8900 |

Table 1. Comparison of Model Accuracies with and without PCA

## References

[1] Z. Ashktorab, C. Brown, M. Nandi, and A. Culotta. Tweedr: Mining twitter to inform, 2014. 1

[2] M. Chaudhari and S. Govilkar. A survey of machine learning techniques for sentiment classification. *International Journal on Computational Science & Applications*, 5(3):13–23, 2015. 1

[3] D. T. Nguyen, K. A. A. Mannai, S. Joty, H. Sajjad, M. Imran, and P. Mitra. Rapid classification of crisis-related data on social networks using convolutional neural networks. *arXiv*, 2016. arXiv:1608.03902. 1