

Objective:

Create a disaster image and text classification application with a web interface and database, allowing users to upload images or tweets manually for classification, and access live classification via a scraper.

My Development Journey in Disaster Detection

1. **Project Start:** Identified the need for a system to classify disaster-related texts and images.
2. **Data Collection:** Gathered disaster tweet and image datasets, addressing class imbalance by merging fine classes into broader categories and weighting minority classes.
3. **Model Development:**
 - **Image Classification:** Implemented a VGG19-based model, tackling performance issues due to class imbalance.
 - **Text Classification:** Built models (Logistic Regression, SVM, etc.) to classify tweets, focusing on balancing sensitivity and specificity.
4. **Real-Time Scraping:** Developed a Twitter scraper using Selenium, overcoming challenges with dynamic data extraction.
5. **Web Application:** Created a Flask app for users to classify data and scrape Twitter, facing integration and user experience challenges.
6. **Evaluation:** Assessed model performance with metrics like accuracy and recall, identifying areas for improvement in both image and text classification.
7. **Future Plans:** Aiming to enhance model accuracy through data augmentation and deeper analysis of misclassifications.