**Social Media-Driven Big Data Analysis for Disaster Situation Awareness**

**-----------------------------------------------------------------------------**

**TEAM**

MANISH – 122AD0029

DINESH – 122AD0041

BALASUBRAMANIAM – 122AD0048

**---------------------------------------------------------------------------------**

**Abstract**

This project explores the use of **Big Data Analytics and AI-driven techniques** to extract disaster-related intelligence from social media platforms like Twitter and Facebook. Traditional disaster response methods (satellite imagery, ground surveys) are often slow and limited in coverage, whereas social media provides **real-time, crowdsourced data**. However, challenges such as **data sparsity, noise, fake news, and lack of geo-tags** complicate analysis.

This research proposes a **distributed processing framework** using **Apache Spark, NLP techniques, and machine learning (Random Forest, DBSCAN clustering, sentiment analysis)** to classify disaster-related tweets, detect high-risk zones, and predict severity levels. The system achieves **78% accuracy in disaster classification** and **99% accuracy in severity prediction**, demonstrating its potential for **real-time emergency response**.

**1. Introduction**

**Problem Statement**

* Disaster response requires **timely and accurate information**, but traditional methods are slow.
* Social media provides **real-time updates**, but analyzing this data is difficult due to:
  + **High volume of irrelevant posts**
  + **Fake news and misinformation**
  + **Lack of geo-tagged data** (only **0.42%–3.17% of tweets**)
  + **Data ambiguity and redundancy**

**Objective**

* Develop a **Big Data framework** to:
  + Filter disaster-related social media posts
  + Extract actionable insights from limited geo-tagged data
  + Enhance disaster management by integrating **social media intelligence with traditional data sources**

**2. Contributions**

1. **Distributed Big Data Processing**
   * Implemented **Apache Spark** for scalable tweet analysis (1 master + 2 slave nodes).
   * Handled **real-time and batch processing** using **Spark Streaming and Hadoop**.
2. **NLP & Text Preprocessing**
   * Removed noise (emojis, URLs, slang) using **NLTK, spaCy, and custom UDFs**.
   * Applied **stemming, lemmatization, and keyword filtering**.
3. **Machine Learning Models**
   * **Random Forest classifier** (78% accuracy in detecting disaster-related tweets).
   * **Severity prediction model** (99% accuracy using sentiment + distress signals).
   * **DBSCAN clustering** for spatial hotspot detection.
4. **Sentiment & Anomaly Detection**
   * Used **VADER and TextBlob** to detect panic/distress signals.
   * **Z-score analysis** to identify fake news and outliers.
5. **Visualization & Dashboard**
   * **Folium maps** for disaster hotspots.
   * **Word clouds, trend graphs, and severity heatmaps**.

**3. Literature Survey**

| **Study** | **Contribution** |
| --- | --- |
| **Sakaki et al. (2010)** | Used Twitter for real-time earthquake detection using Bayesian filters. |
| **Imran et al. (2015) - AIDR** | Automated disaster tweet classification using ML. |
| **Zou et al. (2020) - ST-ResNet** | Combined CNNs and LSTMs for disaster forecasting. |
| **Burel et al. (2021)** | Used **Graph Neural Networks (GNNs)** for multimodal disaster detection. |

**4. Limitations**

1. **Spatio-Temporal Uncertainty**
   * Only **0.42%–3.17% of tweets are geo-tagged** → reliance on **geoparsing**.
2. **Data Ambiguity**
   * Duplicate tweets, misleading content, and sarcasm affect classification.
3. **Scalability Issues**
   * Processing **millions of tweets in real-time** requires high computational power.
4. **Sentiment Ambiguity**
   * Short texts (tweets) make it hard to detect urgency accurately.

**5. Proposed Methodology**

**System Architecture**

1. **Data Ingestion**
   * Sources: **Twitter API, Facebook Graph API, web scraping**.
2. **Preprocessing**
   * Text cleaning, deduplication, spam filtering.
3. **Feature Extraction**
   * TF-IDF, sentiment scores, keyword frequency.
4. **Machine Learning Models**
   * **Classification**: Random Forest (disaster vs. non-disaster).
   * **Clustering**: DBSCAN for hotspot detection.
   * **Sentiment Analysis**: VADER + TextBlob.
5. **Visualization**
   * **Folium maps, Tableau dashboards, word clouds**.

**6. Experimental Results**

**Key Findings**

| **Metric** | **Performance** |
| --- | --- |
| Disaster Classification (Random Forest) | **78% Accuracy** |
| Severity Prediction | **99% Accuracy** |
| Real-Time Earthquake Detection | **85% Accuracy (within 60 sec)** |
| Flood Spread Prediction (ST-ResNet) | **87% Accuracy** |

**Visualizations**

* **Spatial clustering** identified high-risk zones (e.g., flood-prone areas).
* **Sentiment analysis** correlated panic levels with government response delays.

**7. Conclusion & Future Work**

**Conclusion**

* The system successfully **automates disaster detection** using social media.
* **Spark-based processing** ensures scalability for real-time analysis.

**Future Work**

1. **Edge Computing** – Reduce cloud dependency during disasters.
2. **Multilingual Support** – Analyze tweets in regional languages.
3. **Multimodal AI** – Incorporate images/videos for better predictions.
4. **Government Integration** – Link with emergency response systems.

**8. References**

1. Sakaki et al. (2010) – Twitter for earthquake detection.
2. Imran et al. (2015) – AIDR for disaster classification.
3. Zou et al. (2020) – ST-ResNet for spatio-temporal forecasting.
4. IEEE, Springer, MDPI papers on social media analytics.

**Team:**

Manish – 122AD0029

Dinesh, -122AD0041

Balasubramaniam – 122AD0048