Graduation Project Idea and Description

HAST

إشراف

م.

Abstract

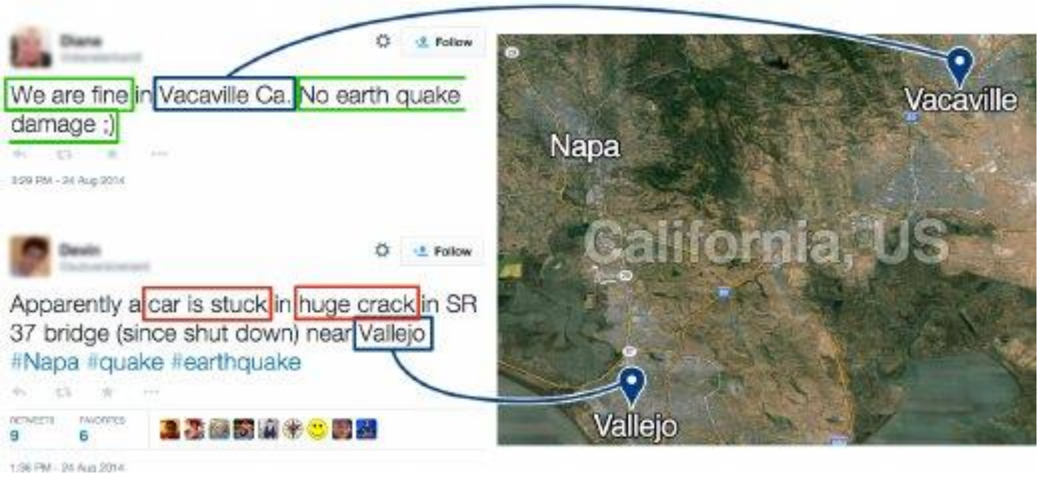
This project aims to develop an AI-driven system for detecting and analyzing local events from social media in real time. By processing text ~~and images~~, the system identifies incidents such as crimes, accidents, and natural disasters. It follows a modular design to ensure scalability and efficiency, enabling real-time data handling and visualization through an interactive web platform. Providing a reliable solution for event monitoring and situational awareness.

Keywords: Event Detection, Contextual Awareness, Crisis, NLP.

Titles:

- **A Deep Learning-Based Framework for Local Event Detection Using Social Media Data.**
- **Deep Learning Framework for Local Event Awareness through Social Media Streams.**
- **Real-Time Emergency Detection System Powered by AI and Social Media Analysis.**
- **Automated Social Media Event Detection and Visualization for Crisis Awareness.**
- **Event-Radar: Real-time Local Event Detection System for Geo-Tagged Tweet Streams**
- **An AI-Powered Emergency Alarming System.**
- **Leveraging Deep Learning Techniques and Social Media Streams to Achieve Local Events Awareness.**

Summary:



We are fine in Vacaville Ca. No earth quake damage ;)

3:29 PM - 24 Aug 2014

Apparently a car is stuck in huge crack in SR 37 bridge (since shut down) near Vallejo #Napa #quake #earthquake

RETWEETS 9 FAVORITES 6

1:56 PM - 24 Aug 2014

Objectives:

1. **Develop an AI-Powered Event Detection System:** Build a scalable and reliable system using deep learning techniques to analyze social media data for detecting local events (e.g., crimes, accidents, and natural disasters).

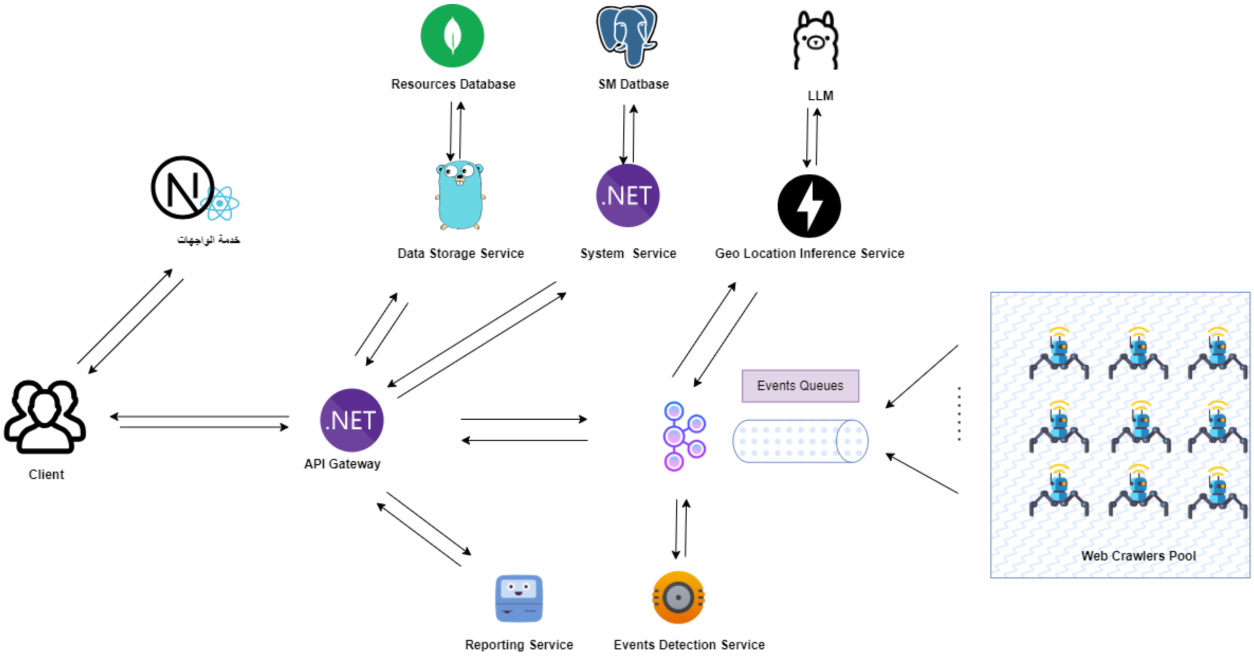
2. **Ensure Software Engineering Best Practices:** Apply software engineering principles such as modular design, maintainability, scalability, and system performance optimization.

3. **Design a User-Centric Application:** Develop a web-based interface for visualizing real-time event data, ensuring usability, accessibility, and responsiveness.

4. **Optimize Data Processing:** Efficiently handle large-scale social media data streams, ensuring both real-time processing and historical data analysis.

5. **Evaluate and Validate the System:** Assess both the AI model's accuracy and the software system's performance, ensuring robustness and reliability.
6. **Deploy and Document the System**: Deploy the system in a real-world environment, ensuring its stability, and thoroughly document the development process, the AI models, the system's architecture, and performance evaluations.
7.

System Architecture Overview:



Main Deep Learning Tasks:

1. **Informative Tweet Detection:**
   - Identify and classify social media posts that report local events using advanced transformer-based models.
2. **Event Detection and Entity Extraction:**
   - Identify the type of event within informative tweets and extract key entities
   - Extract key named entities (e.g., people, organizations, locations) from both text and images to support event classification and situational awareness
3. **Geo-Location Extraction and Inference:**
   - Extract geo-references from text
   - Perform geocoding to map events, and infer locations for non-geotagged posts.
   - This can include determining the most probable location based on context and language.
4. **Geospatial Question Answering:**
   - If time permits, implement a geo-spatial question-answering system
   - That enables users to ask location-based queries about events (e.g., "Where did the earthquake occur?" or "What were the nearest landmarks to the fire?") based on social media data.
5. ~~**Sentiment and Urgency Analysis:**~~
   - ~~Analyze the emotional tone and urgency of posts to assess the severity and relevance of events.~~
6. ~~**Image Classification for Event Identification:**~~
   - ~~Use CNNs to classify event-related images and identify visual evidence of incidents.~~
7. ~~**Multimodal Event Integration:**~~
   - ~~Combine textual and visual information using multimodal deep learning techniques to enhance event recognition and contextual understanding.~~

Methodology:

1. Software Engineering Process:

a. **Software Development Methodology:** Adopt an **Agile Methodology** with iterative development and continuous feedback loops to ensure flexibility and adaptability.

b. **Service-Oriented Architecture (SOA):**

Use a **Service-Oriented Architecture (SOA)** to design the system as a collection of loosely coupled, reusable, and independently deployable services that communicate over well-defined interfaces.

Each service is focused on a specific business capability, ensuring the system is scalable, maintainable, and can be independently updated without disrupting the whole system.

The core services include:

- **Web Crawling Service:**
  - **Crawlers** are used to extract relevant social media posts from public pages, websites, and social platforms.
  - *Social Media Crawler* to crawl websites, blogs, or news articles for event-related content, leveraging APIs and web scraping techniques to gather posts.
  - *Platform-Specific Crawler* to handle crawling specific platforms (e.g., Twitter, Facebook, Instagram) using platform APIs to fetch posts, comments, and hashtags related to local events.
  - **Kafka Queuing:** Crawlers send the raw collected posts and data into Kafka queues, which allow them to be processed by downstream services such as **Preprocessing Service** or **Event Detection Engine**.
- **Data Ingestion Service:**
  - Collects and ingests social media posts using APIs or datasets from crawlers.
  - Process real-time data streams from crawlers and push them into Kafka for further processing.
- **Preprocessing Service:**
  - Handles data cleaning and preparation using NLP techniques. This service prepares data for event detection and analysis.
  - **New Service:** *Named Entity Recognition (NER) Service* to extract key entities such as locations, people, and events from the text.
- **Event Detection Service:**
  - Runs deep learning models to classify social media posts and identify whether they report events (e.g., crime, accidents, natural disasters).
- **Geo-Location Inference Service:**

  - Identifies and geocodes locations mentioned in the posts, including non-geotagged posts by inferring location context from text. This service uses geocoding APIs and NLP-based location inference.
- **Geo-Spatial Question Answering (QA) Service:**
  - Answer location-based queries and provide additional insights based on geo-referenced data.
- **Event Classification and Entity Extraction Service:**
  - Extracts event types (e.g., accident, protest, fire) and additional entities (e.g., people, organizations) to provide detailed event descriptions.
- **Visualization Dashboard Service:**
  - Displays the detected events on an interactive map in real-time, allowing users to explore incidents and view detailed event reports. The dashboard updates dynamically with new events as they are detected.
- **Events Reporting Service:**
  - Generate daily summaries and reports on the detected events and provide insights on trends over time.
- **Storage Service:**
  - Responsible for storing the processed data, including social media posts, event information, entity data, and geolocation data.
  - The service uses a **NoSQL database** to store structured and unstructured data at scale.

c. **Queuing with Kafka:**

- Use **Apache Kafka** for asynchronous communication and queuing between the services.
- **Data Ingestion Service** sends social media posts to Kafka topics for further processing.
- **Preprocessing Service** subscribes to Kafka topics, processes data, and passes it to the **Event Detection Service** via Kafka.
- The **Geo-Location Service**, **Event Classification Service**, and other services are connected through Kafka, which ensures reliable message delivery, handles backpressure, and scales horizontally.

d. **Testing and Validation:**

- Conduct **unit testing** and **integration testing** to ensure system reliability.
- Use **load testing** tools to evaluate system performance under high data loads.

2. AI Model Development:

The AI system will be designed to process and analyze both textual ~~and visual data~~ for accurate and efficient event detection from social media. The development consists of multiple machine learning tasks that work together to detect and classify events, extract relevant details, and analyze contextual information.

- Tweets and Messages Analysis

The system processes social media posts to extract meaningful information and classify events.

- **Informative tweet classification:** determines whether a post contains relevant event information. the system distinguishes between general discussions and actual event reports.
- **Event type detection:** identifies the category of an event (e.g., accidents, crimes, natural disasters) based on the content of the post.
- **Entity extraction:** extracts essential details such as location, time, people involved, and event descriptions.
- **Geo Location Inference:** extracts and processes location-related information from posts to place events on a map.

- Geo-location extraction

Events must be mapped accurately for effective detection and response. The system extracts and processes location-related information from social media posts.

- **Text-based location detection:** identifies locations mentioned in posts (e.g., city names, street addresses, or landmarks).
- **Geocoding**: converts extracted locations into precise latitude and longitude coordinates to place them on a map.
- **Contextual disambiguation:** if multiple locations are mentioned, the system determines the most relevant one based on context.

3. Real-Time Data Processing:

e. Use **Kafka** or **RabbitMQ** for message queuing to handle real-time social media streams.

f. Implement parallel processing using **multi-threading** or **asynchronous programming** to ensure low-latency event detection.

References:

Task - Event Detection:

[1.] Distilbert-gnn: a Powerful Approach to Social Media Event Detection, 03 April 2024, PREPRINT (Version 1) available at [https://doi.org/10.21203/rs.3.rs-4193412/v1]

[2.] A Survey on Event Tracking in Social Media Data Streams, vol. 7, no. 1, pp. 217-243,(2024), doi: 10.26599/BDMA.2023.9020021.

[3.] DAMe: Personalized Federated Social Event Detection with Dual Aggregation Mechanism. (2024). 3052-3062. 10.1145/3627673.3679551.

[4.] Real-time event detection in social media streams through semantic analysis of noisy terms. J Big Data 9, 90 (2022). https://doi.org/10.1186/s40537-022-00642-y

[5.] Deep-Eware: spatio-temporal social event detection using a hybrid learning model. J Big Data 9, 86 (2022). https://doi.org/10.1186/s40537-022-00636-w

[6.] Fine-grained location prediction of non geo-tagged tweets: a multi-view learning approach. (2022). 82-91. 10.1145/3557918.3565875.

[7.] Real-time event detection in social media streams through semantic analysis of noisy terms. Journal of Big Data. (2022). 9. 10.1186/s40537-022-00642-y.

[8.] Event Detection from Social Media Stream: Methods, Datasets and Opportunities, (2022). doi: 10.1109/BigData55660.2022.10020411.

[9.] Suspicious Local Event Detection in Social Media and Remote Sensing: Towards a Geosocial Dataset Construction. (2020). 10.1109/ATSIP49331.2020.9231798.

[10.] Event-Radar: Real-time Local Event Detection System for Geo-Tagged Tweet Streams. (2017).

Task - Geolocation Inference:

[1.] Extracting geo-references from social media text using bi-long short term memory networks. (2024).

[2.] Predicting the Geolocation of Tweets Using transformer models on Customized Data. (2024).

[3.] Leveraging Large Language Models to Geolocate Linguistic Variations in Social Media Posts. (2024).

[4.] Fine-Grained Location Prediction of non Geo-tagged Tweets - A Multi-view Learning Approach. (2024).

[5.] IDRISI-D: Arabic and English Datasets and Benchmarks for Location Mention Disambiguation over Disaster Microblogs. In *Proceedings of ArabicNLP 2023*, pages 158–169, Singapore (Hybrid).

[6.] A Named Entity Recognition and Topic Modeling-based Solution for Locating and Better Assessment of Natural Disasters in Social Media. (2024).

[7.] Location Name Extraction from Targeted Text Streams using Gazetteer-based Statistical Language Models. (2020).

[8.]

Task - Platform Design:

[1.] Sensing Real-World Events Using Arabic Twitter Posts. (2025).

[2.] Designing a Prototype Platform for Real-Time Event Extraction: A Scalable Natural Language Processing and Data Mining Approach. (2024).

Task – Event Extraction:

[1.] Towards Event Extraction with Massive Types: LLM-based Collaborative Annotation and Partitioning Extraction. (2025).

[2.] Fine-Grained Meetup Events Extraction Through Context-Aware Event Argument Positioning and Recognition. (2024).

Task – Crisis Event Detection:

[1.] A Social Context-aware Graph-based Multimodal Attentive Learning Framework for Disaster Content Classification during Emergencies. Expert Systems with Applications. (2024). 10.1016/j.eswa.2024.125337.

[2.] Kawārith: an Arabic Twitter Corpus for Crisis Events. (2021).

[3.] Enhanced Arabic disaster data classification using domain adaptation. PLOS ONE (2024). 19(4): e0301255. https://doi.org/10.1371/journal.pone.0301255

[4.] Zero-Shot Classification of Crisis Tweets Using Instruction-Finetuned Large Language Models. (2024). 10.48550/arXiv.2410.00182.

[5.]FloDusTA: Saudi Tweets Dataset for Flood, Dust Storm, and Traffic Accident Events. (2020).

[6.]

Datasets:

[1.] Kawarith:

- An Arabic Twitter Corpus for Crisis Events, It Comprises Arabic tweets from 22 crisis events that occurred between October 2018 and September 2020.
- Unlabelled Corpus: A large-scale crisis-related Arabic Twitter corpus of 1,658,795 unique tweets from 22 emergency events.
- Labelled Data: A gold-standard dataset comprising ~12k unique tweets from seven events: the Jordan floods, Kuwait floods-18, Hafr Albatin floods-19, the Cairo bombing, the Dragon storms, the Beirut explosion and Covid-19. Apart from Covid-19, which was labelled by relatedness to the event, tweets were annotated in terms of information type in a multi-label schem.

[2.] IDRIS

- IDRISI is the largest-scale publicly-available Twitter Location Mention Prediction (LMP) dataset, in both English and Arabic languages.
- Contains 26 disaster events of different types (e.g., floods, earthquakes, fires, etc.) that occurred in a wide geographical area of the English- and Arabic-speaking countries across continents.

[3.] FloDusTA

- The dataset contains tweets written in both Modern Standard Arabic (MSA) and Saudi dialect for the purpose of detecting flood, dust storm and traffic Accident.

[4.] TSEqD (Turkey-Syria Earthquake Dataset)

- Multimodal dataset is a comprehensive collection of manually annotated comprising 10,352 tweets and associated images obtained during the period from February 6, 2023, to March 16, 2023.
- Annotated for two distinct tasks: Informative Tasks and Humanitarian Tasks.
- The process of data curation, annotation, and label distribution across various classes is elaborated in subsequent sections, ensuring a comprehensive understanding of the dataset's composition and utility.

[5.] EveTAR

-

[6.] MAVEN

- A Massive General Domain Event Detection Dataset
- Contains 4,480 Wikipedia documents, 118,732 event mention instances, and 168 event types.

[7.] ACE 2005 Multilingual Training Corpus

- Multilingual Training Corpus contains the complete set of English, Arabic and Chinese training data for the 2005 Automatic Content Extraction (ACE) technology evaluation.
- The corpus consists of data of various types annotated for entities, relations and events by the Linguistic Data Consortium (LDC) with support from the ACE Program and additional assistance from LDC.

Project Timeline:

Phase	Tasks	Duration
Phase 1: Literature Review	Literature review, defining objectives, selecting datasets	2 weeks
Phase 2: Theoretical Study	Study foundational concepts of deep learning, NLP, and computer vision	1 week
Phase 3: Requirement Analysis	Define system requirements, technical specifications.	1 week
Phase 4: Data Collection & Preprocessing	Collect datasets and apply preprocessing techniques.	1 week
Phase 5: Deep Learning Model Development	Train transformer-based models for text and image classification.	3 weeks
Phase 6: Event Detection System Piplines	Integrate models into an event detection pipeline.	1/2 week
Phase 7: Microservices Development	Implement backend microservices, APIs, and communication mechanisms	3 weeks
Phase 8: Frontend System Development	Develop the user interface and dashboard	2 weeks
Phase 9: Real-Time Processing	Implement real-time data processing using Kafka or RabbitMQ.	1/2 week
Phase 10: Performance Evaluation	Evaluate AI models assess system performance.	1/2 week
Phase 11: Deployment	Deploy the system.	1/2 week
Phase 12: Documentation & Final Report	Document the entire development process, DL tasks, Resullts.	1 week

Total Duration: 16 weeks (approximately 4 months)