Introduction To Information Retrieval CSE 535

Project 4
Team : Jarvis

Submitted By:

Vishal Raman
Vishwas Nalka
Carolyn Bose
Pushkar Pandey

Introduction:

This project aims to create a tweet search engine using Apache Solr and is a part of the UB CSE535 Information Retrieval course. The project's goal is to successfully crawl tweets from Twitter, index them, and create a user interface that allows users to search tweets. Furthermore, the project contributes to analyzing the corpus and showing insights to the user about the impact of the tweets in the twitter sphere. The users can search words, phrases, hashtags etc. and apply filters by POI, language and country. It offers a rich, flexible set of features for search, like query boosting and finding disinformation tweets. Furthermore, the search engine allows the user to search in multiple languages.

Methodology:

1) Front-end:

Used **Angular** 13, Typescript-based web framework for hosting the system along with Angular Material for responsive modules. For data representation and visualization, Highcharts and chart.js which are flexible charting libraries were used.

- We displayed data for both the complete corpus of 100587 tweets and for dynamic data depending on the query and 3 filters- POI, language, country. We also displayed data by filtering disinformation.
- Implemented bar graphs, doughnut charts, pie charts, radar charts for data visualization.
- Word cloud for topic relatedness was implemented for representing related topics for each query.

2) Back-end:

- **Tweets and Replies indexed:**
  Overall 100587 tweets were collected that covered all requirements mentioned in Project 1 (POI, Non-POI, Reply tweets and so on). There were 17837 POI related tweets and 23 POIs in the dataset. And 33060 reply tweets were collected using the tweet IDs of other tweets in the dataset.

- **Query parser:**
  The search queries from the UI were processed based on the learnings from Project 3.
  1. Every query is parsed and the hashtags were extracted from it. If the query had hashtags, this was added as search terms on the 'hashtags' field in the index with higher boosting than other fields.
  2. Language detection was performed on each query and based on the detected language, the query was translated to other two languages. Translated text in each language was queried against 'text_en', 'text_hi', 'text_es' respectively with relatively high boosting.
  3. Finally, the original query is also searched in 'tweet_text' and 'hashtags' fields with very low boosting to improve the recall of the system.

- **Language Translation:**

Spacy package's **spacy_langdetect** *(LanguageDetector)* was used to detect the language and confidence of the detection. If the confidence score was higher than 95%, language translation was done. For language translation, the Translator API from **googletrans** was utilized.

- **Topic-modelling:**
  This was done by finding the Tf-idf scores of all the words in tweets related to the query. **CountVectorizer** was used first with stopwords removed (English, Hindi and Spanish) to vectorise the tweets. Then **TfidfTransformer** was used on the CountVectorized data to get Tf-idf scores for each word. Then the top 20 words with the highest Tf-idf score were returned and displayed as a wordcloud. This supported multi-lingual work clouds (i.e. word cloud in hindi, english and spanish)

- **Vaccine Dis-information:**
  Training data samples (Around 50 sentences) were collected initially that contained information regarding misinformed facts about the covid vaccine. This set of training data was used in a Bert based sent2vec to get the sentence embeddings of these training data and the actual tweets present in the index. Then cosine similarity between the embeddings of the tweets and the embeddings of the training data were computed. Using a threshold of 0.7, most similar tweets w.r.t the training data were re-indexed into a separate SOLR core meant for searching this Disinformation data. Overall 2371 tweets in the original tweet dataset were found as disinformation.

- **Vaccine hesitancy:**
  A few queries related to vaccine hesitancy were formulated, and the queries were used to fetch tweets related to vaccine hesitancy. The query parser that we have built, searches for the query in every field indexed, and this way the parser was resilient enough to get relevant tweets w.r.t vaccine hesitancy.

- **Sentiment Analysis:**
  To perform sentiment analysis, the TextBlob package was used. For each tweet, sentiment score and the polarity were computed. The 3 buckets of polarity were used viz. positive, negative & neutral. For every tweet, the computed sentiment score and the polarity result were indexed into SOLR.

- **Voice to text:**
  SpeechRecognition package was used to capture the voice of the user using the device's microphone. The voice was then rendered and transferred to text using **recognize_google** which is a part of the SpeechRecognition package. Then the "country" and "language" were extracted from the text (country and language are the filters), and the remaining part of the text was chosen as the "query"

Sample screenshots:

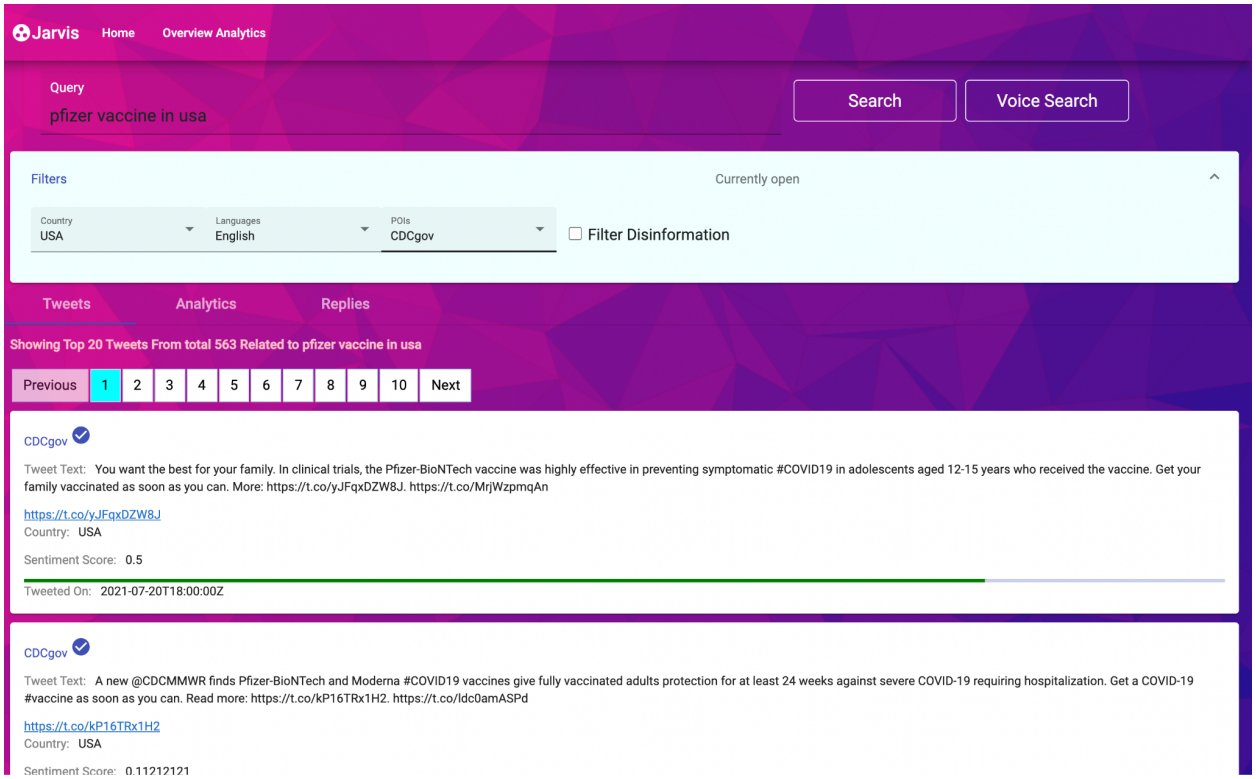

fig - Search and filter Page

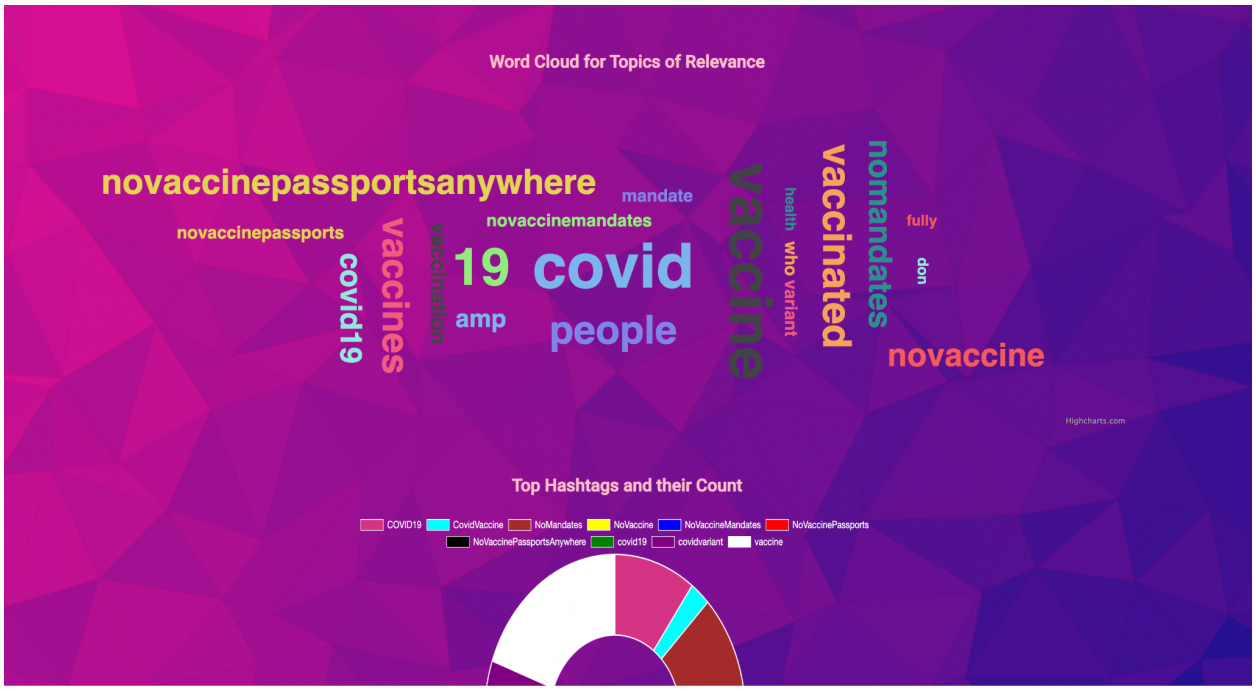

fig - Word Cloud and Top Hashtags

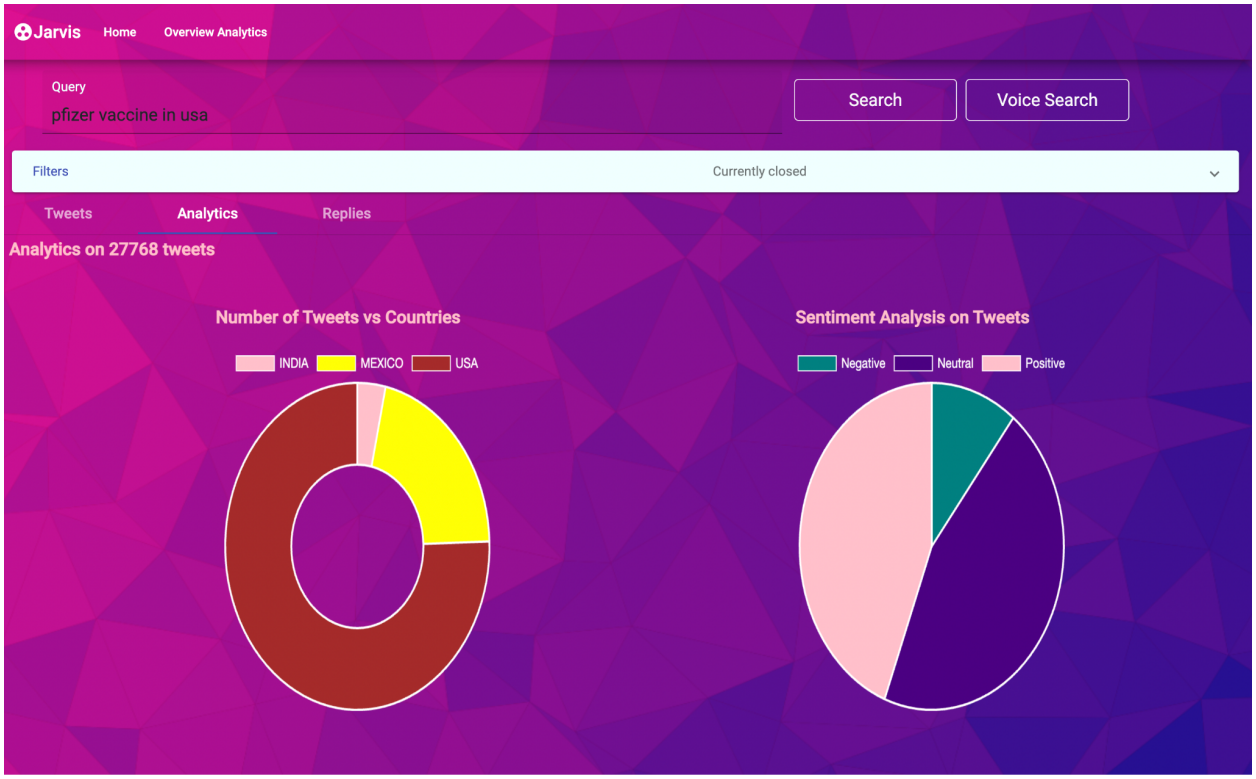

fig - Analytics Page

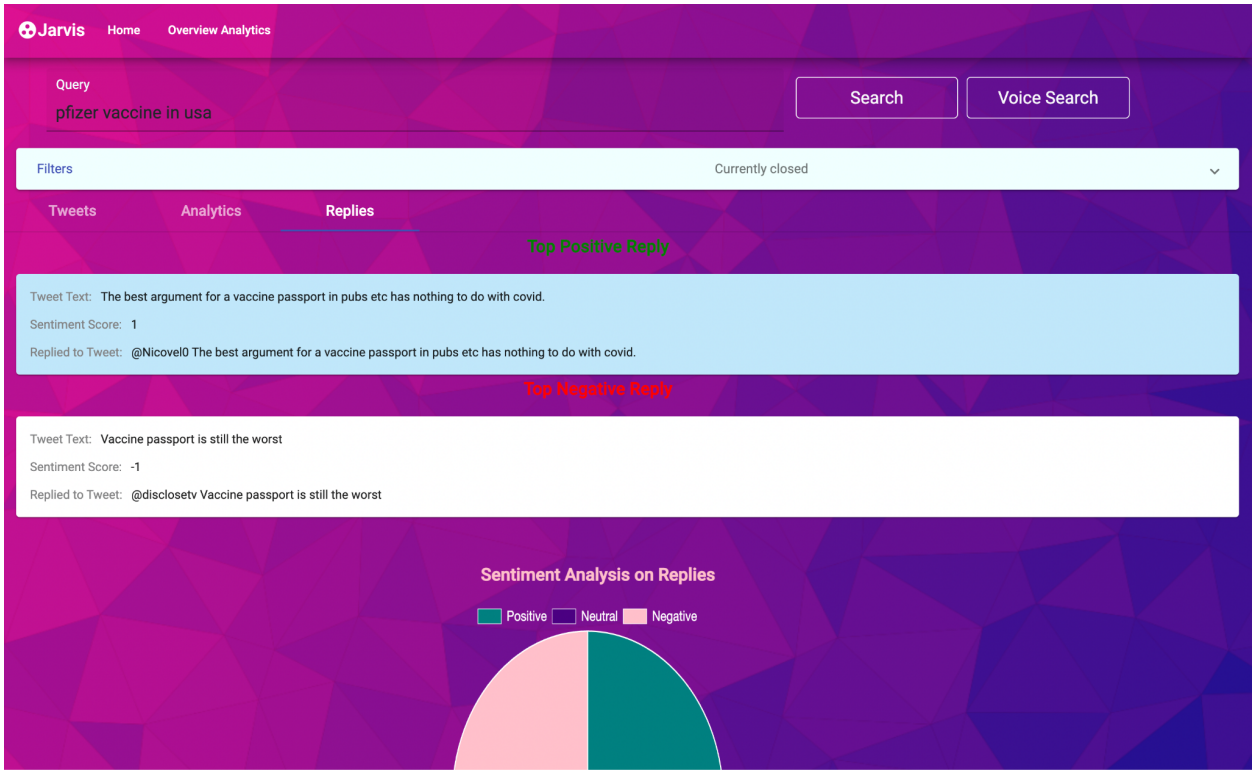

fig - Replies Page

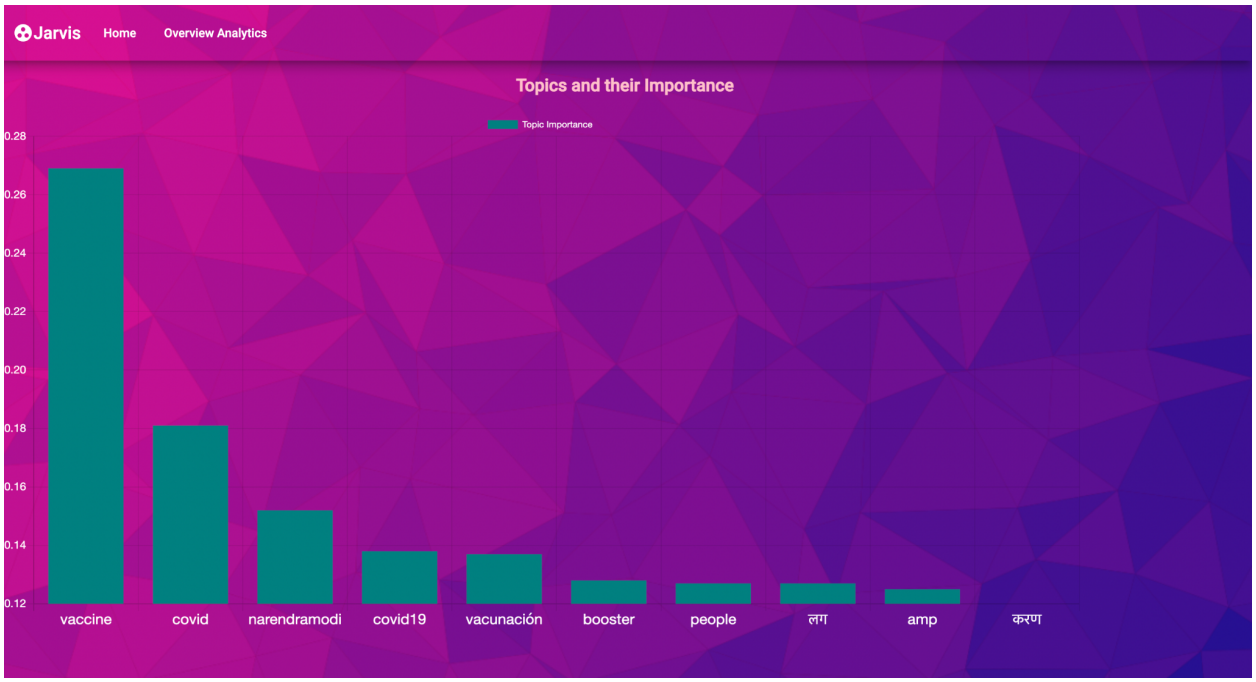

fig - Overview Analytics

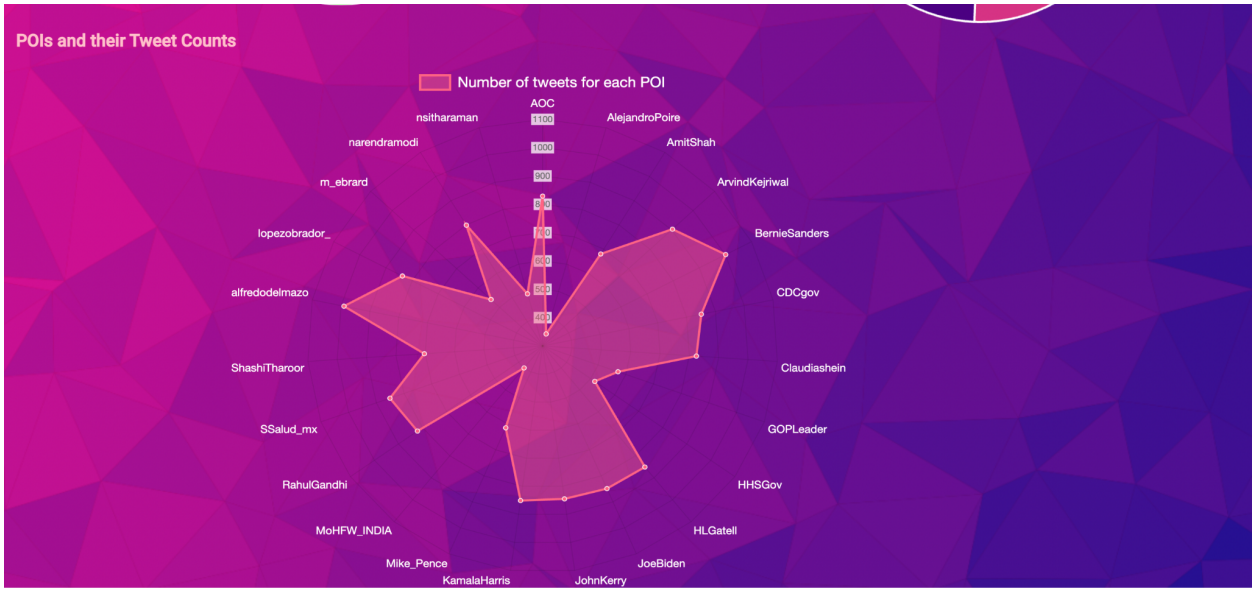

fig - Radar chart

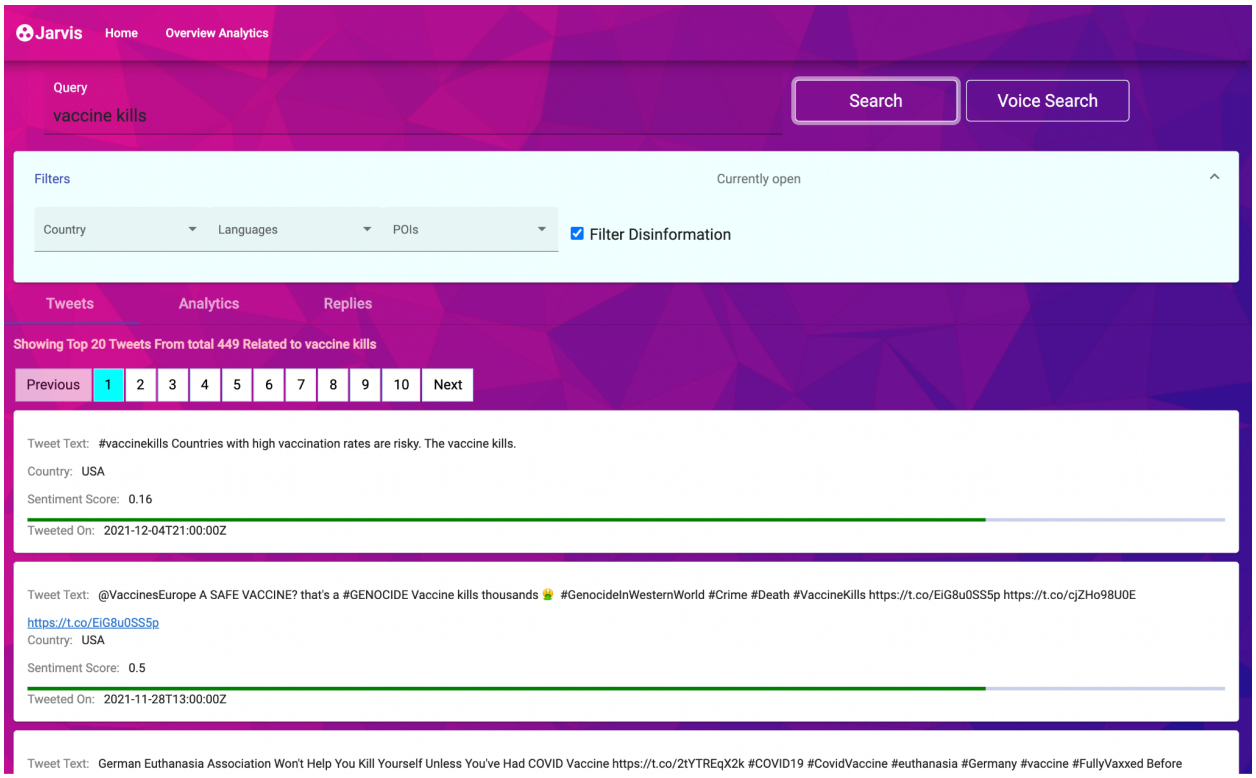

fig - Disinformation Tweets filter

Conclusion:

1. Based on the learning outcomes from the first 3 projects, end-to-end search engine was implemented to analyse tweets dataset. Various aspects such as collecting required data (tweets & replies), indexing the data and augmented information (sentiment, disinformation, hesitancy related data) in SOLR were done. Query reformulation and boosting were implemented to fetch relevant tweets for the user queries and improve the overall recall.
2. Use cases such as Topic Modelling, Sentiment Analysis, Content Analysis (Static and Dynamic charts and visualizations), Vaccine Hesitancy, Vaccine Disinformation, Voice Search (voice to text with filters), Language Detection and Translation were implemented as well.
3. All the above features were exposed to the user in an interactive UI. Overall analytics of the indexed tweets were displayed in the UI.
4. Future Scope:
    a. Stance detection and persuasion for or against taking the vaccine could have been implemented and shown in a better manner.
    b. Visualization based on date to show POI activity across date ranges can be shown.
    c. Highlighting the user query matches in the result could have been done to enhance the user experience.
    d. Related news articles/Summarization Tab based on query could be implemented to give more information to the user.