CSE 535: INFORMATION RETRIEVAL

PROJECT 4: Dissecting Twitter data to analyze government & public attitude towards Covid and vaccines

Team name: **Archie**

Project done by:

NAME	UB PERSON ID:	UB IT NAME:
Sivakumar Pasupathi	50366350	spasupat
Parthiban Rajendran	50415960	rajendr3
Vaishnavi Ruppa Gangadharan	50418483	vruppaga

Introduction

The user interface has been designed to develop an efficient search engine in such a way that the user can type the input query in the search box and the data collected is shown in the web page. The tweets related to the query, the name of the user who posted that particular tweet, the sentiment score for each tweet, an option to redirect to the user's page.
In addition to this, there are options for applying advanced filters based on Person of Interest (POI), Country and Language. Graphs based on the collected data are plotted in order to give a better visual representation.

Methodology

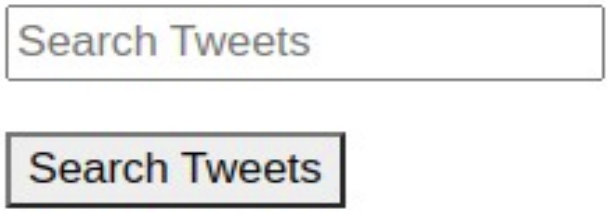
Technologies used

Frontend: HTML, CSS, Javascript, Bootstrap, jQuery, AJAX
Libraries used for Visualization: Chart.js
Backend: Flask application
IR system: Solr

User Interface

A text box is provided for the user to type the input queries to perform the search operation. Once the user clicks the search button, the system creates a http request to the Solr instance to fetch the data. The http response received from Solr is a list of JSON objects which are processed and displayed in the UI.

Figure 1: Input search box and Search button



The top ten results are displayed on the first page and an extra option is also enabled in order to navigate to the next pages which will retrieve the next top ten results.

Figure 2: Search results

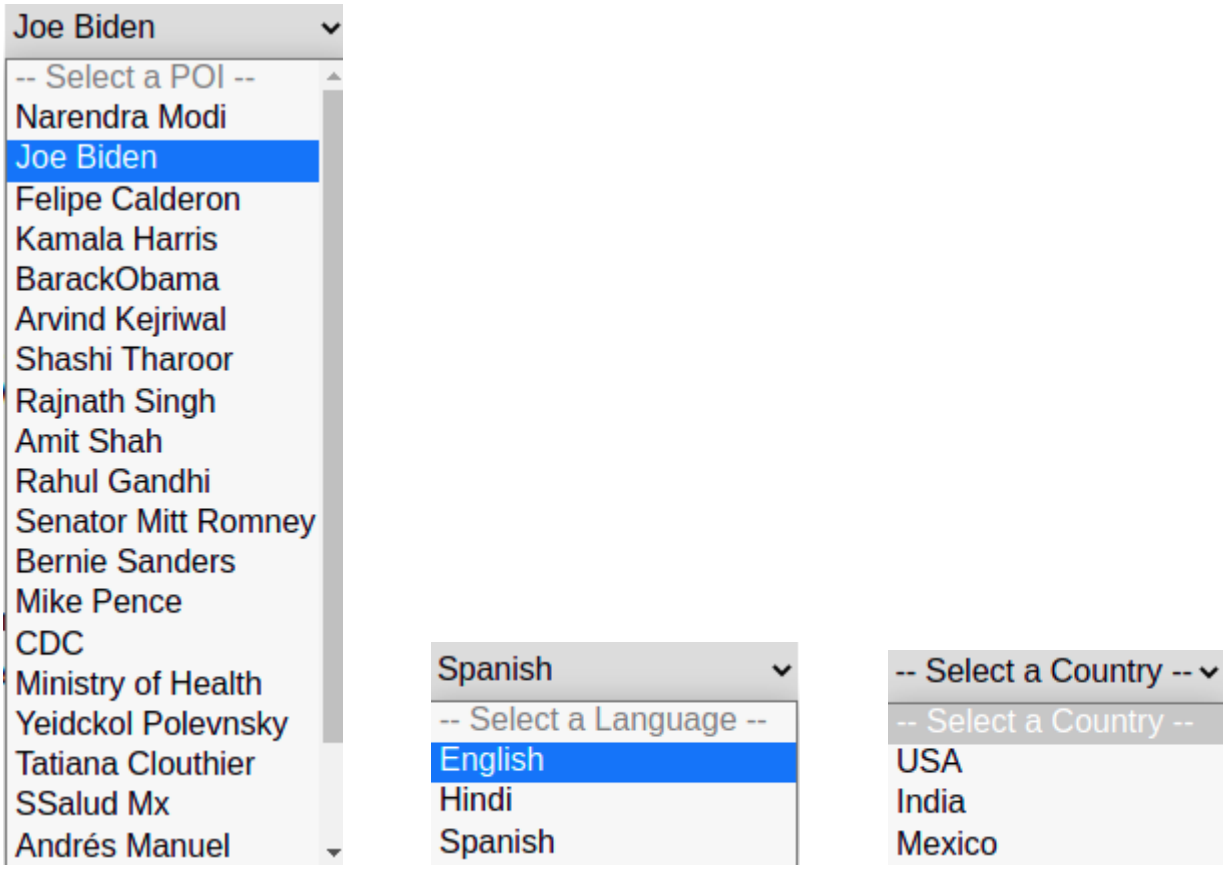
Advanced Search

Additional filtering options are provided as dropdowns to the user.
The various filters which are available are:

- Person of Interest
- Country
- Language

The search query is formed based on the filtering options selected by the user and a request is sent to the Solr instance to retrieve the top 10 relevant tweets.

Figure 3: Advanced filters

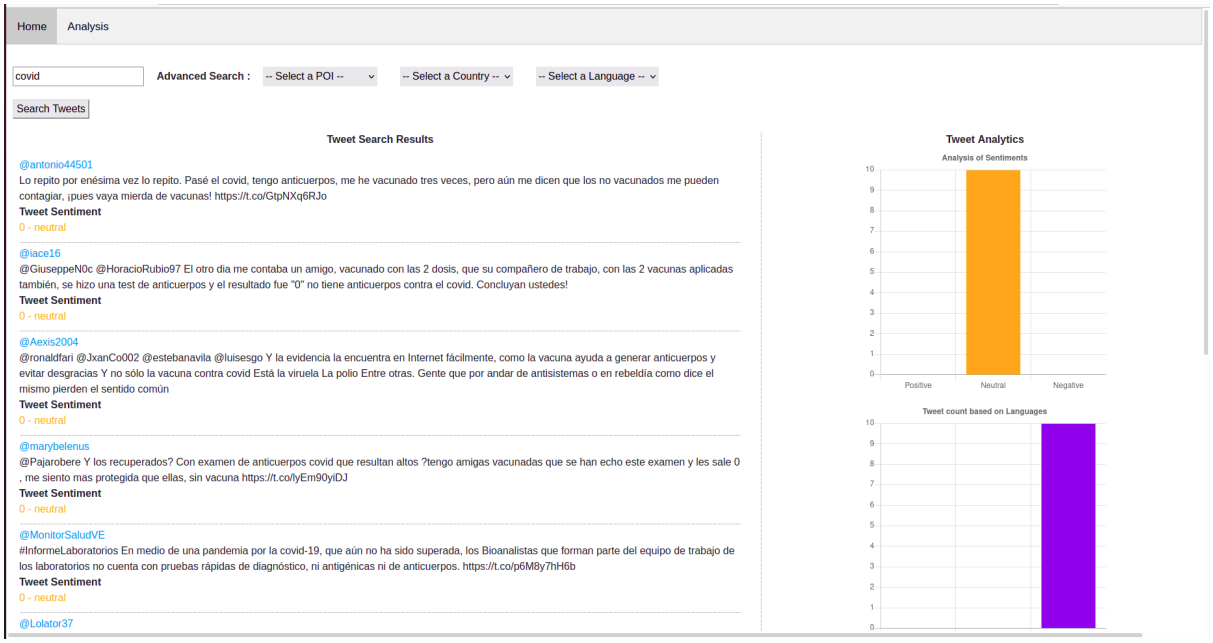


Search results

The sample results are shown in the screenshot below.
The contents which are displayed on the web page contains the following details:

- the tweets related to the query
- the name of the user who posted that particular tweet/ POI name
- the sentiment score for each tweet
- an option to redirect to the user's page.
- an option to redirect to the next pages

Figure 4: Search results



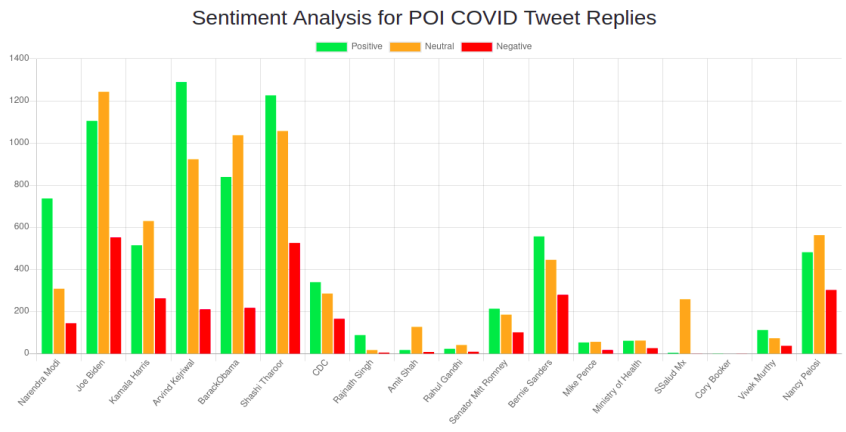
Charts

Chart.js is a free open-source JavaScript library for data visualization, which supports various chart types: bar, line, pie etc. This library has been used to display the graphs based on the collected data.
A sample graph which is plotted using Chart.js in the UI is shown below.

Sentimental analysis

The sentimental score has been calculated for each tweet using **textblob** library and a graph has been plotted to visualize the data for each POI's tweets.

Figure 5: Sample graph to analyse the sentimental score



Information displayed

The number of results displayed per page is restricted to 10. This number is configurable in the backend. To view more results, an option has been provided to navigate to the other pages and view the results.

Figure 6: View more results option

Data retrieval

The data has been collected from twitter by using tweepy API. A script is designed to crawl the tweet data and effectively index the data into Solr. The data fetched using the script satisfies various conditions provided which include total number of tweets (50k), gathering data for particular POI (500 tweets per POI for 5 POIs per country), gathering the reply tweets for covid vaccine related tweets and covid related POI tweets.
This dataset is used for developing a multi-lingual IR system.

Processing of tweets

The data collected has more information related to the tweets. Our script only retrieves the required fields and writes the data into a json file. The date field in the collected data is not in the format which can be indexed into Solr. So, minor modifications are done using python libraries to format the fields.

List of fields

Id, country, tweet_lang, tweet_text, tweet_date, verified, hashtags, mentions, tweet_urls, poi_id, poi_name, text_en, text_hi, text_es, tweet_emoticons - These are the list of the fields for which we index the data into Solr.

For collecting replies, there are two additional fields which are indexed, namely, reply_to_tweet_id,reply_to_user_id,reply_text.

Model used for indexing

Okapi BM25 (BM - best matching) is a ranking function used by search engines to estimate the relevance of documents to a given search query. It is based on the probabilistic retrieval framework. This model has been used for indexing the data into Solr. The values of hyperparameters are b= 0.75 and k=1.2.

Relevance using BM25 model

Tokenizers and filters which are used while indexing the data into Solr include StandardTokenizerFactory, SynonymGraphFilterFactory, StopFilterFactory, LowerCaseFilterFactory, EnglishPossessiveFilterFactory, KeywordMarkerFilterFactory, PorterStemFilterFactory.
This improves the performance of the search engine by providing the most relevant tweets.

Figure 7 : BM25

```
    {
        'name': 'text_en',
        'class': 'solr.TextField',
        'positionIncrementGap': '100',
        'indexAnalyzer': {
            'tokenizer': {
                'class': 'solr.StandardTokenizerFactory'
            },
            'filters': [{
                'class': 'solr.StopFilterFactory',
                'words': 'lang/stopwords_en.txt',
                'ignoreCase': 'true'
            }, {
                'class': 'solr.LowerCaseFilterFactory'
            }, {
                'class': 'solr.EnglishPossessiveFilterFactory'
            }, {
                'class': 'solr.KeywordMarkerFilterFactory',
                'protected': 'protwords.txt'
            }, {
                'class': 'solr.PorterStemFilterFactory'
            }, {
                'class': 'solr.EnglishPossessiveFilterFactory'
            }]
        },
        'similarity': {
```

Work distribution

Sivakumar Pasupathi - Web application, Tweet collection, Analysis of data
Parthiban Rajendran - Tweet collection, Integration, Analysis of data
Vaishnavi Ruppa Gangadharan - Visualization, Report and Analysis of data

Conclusions, Results and Analysis

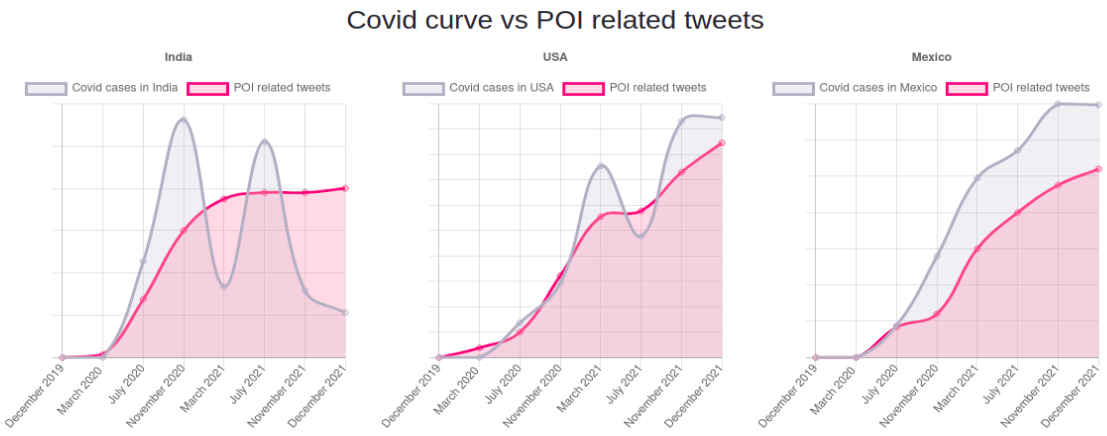
The analysis is performed by providing various queries as input to the search engine and it is observed that the most relevant tweets are returned for the input query which is posted.

The overall analysis is shown in the second tab of the web search engine which includes 11 visual representations designed based on the collected data.

- Tweets Distribution based on Country and Language
- Number of Covid and Vaccine related tweets
- POI Total Tweet and Covid related tweets count
- Replies Count for each POI
- Covid curve vs POI related tweets
- Sentiment Analysis for POI COVID Tweet Replies
- Attitude of General Population Towards Vaccine
- Vaccine Hesitancy on General Population

Relation between Covid curve and POI related tweets:
From the graphs, it is evident that as the active covid cases started increasing in each country, the tweets tweeted by the POIs also gradually increased.

Figure 8 : Covid curve and POI related tweets



The tweet count distributions for each country (USA, India, Mexico) and language (English, Hindi, Spanish) are also shown in the form of pie charts in the analytics page.
The total tweet counts, tweet reply counts, covid related tweets for each POI is also visually represented.
Finally, a chart is drawn to determine the topic to which the tweets are majorly related.