CSE 535: Information Retrieval Project 4

Analyzing the impact of political rhetoric in traditional and social media

Anushree Parmar (aparmar5@buffalo.edu)

Mansi Shetty (mansikar@buffalo.edu)

Mohd Ehtesham Shareef (mohdehte@buffalo.edu)

Sneha Panicker (spanicke@buffalo.edu)

Table of Contents

Introduction

This project involves analysing the impact of political rhetoric if influential persons of interest (POI) by monitoring their twitter data. To achieve this, we build an end to end IR solution involving content ingestion, search, topic categorization, analytics and visualization. From Project 1, we have a corpus of about 300,000 multilingual tweets from 15 POI's and 3 countries to draw meaningful insights from. Project 2 taught us the algorithms behind query processing where we implemented

DAAT AND and OR. Project 3 involved implementing the different similarity models on SOLR like BM-25, DFR and LM. Project 4 is an amalgamation of the first three projects, along with designing a full fledged web search engine which will help us analyse the twitter data.

Implementation

Data Preprocessing

To get the sentiments and topic of the tweets, the data had to be preprocessed. The tweets were distributed across three languages - English(en), Hindi(hi) and Portuguese(pt). To get the sentiments, all the tweets other than the english language tweets had to be translated to english first.

For language translation, we used the Google Cloud Platform's translation API. Due to the limited credit and huge corpus of tweets, a script was created which went through the tweets one by one and translated only those in which the "tweet_lang" field was not English(en). A field called "translated" was added to the tweet object, which contains the translated text.

Sentiment Analysis

After preprocessing the data, we used the TextBlob library to find out the sentiments of the individual tweet. Upon calling the function to find out the sentiment, it returns the polarity and subjectivity of the text. The polarity is a floating point value in the range [-1,1]. A value of 0 represents a neutral sentiment, a value greater than 0 represents a positive sentiment and a value less than 0 represents a negative sentiment. We used this methodology to assign sentiments to the individual tweets in the corpus.


```
"tweet_text":["@narendramodi हर हर मोदी, घर घर मोदी !!"],
"tweet_lang":["hi"],
"text_hi":["हर हर मोदी घर घर मोदी "],
"mentions":["narendramodi"],
"tweet_date":["2019-09-06T15:00:00Z"],
"subjectivity":[0.0],
"polarity":[0.0],
"sentiment":["Neutral"],
"translated":["Hail Modi, Modi in every house"],
```


Topic Modeling

The corpus was split into 3 parts, corresponding to the 3 countries and the tweets from the POI's of these were extracted. These 3 parts are used to model the topics for the three countries. After removing the stop words and cleaning the data a little more, a bag of words for each topic was obtained using Latent Dirichlet Allocation (LDA). For each country, we get 5 bag of words list, from

which we interpret and get 5 topics. Each of the tweets are then mapped to the closest topic from the 5 respective topics of the country.

SOLR

We experimented with different similarity models and found out that BM-25 model gives the best recall rate. Upon trying out various combinations for the two parameters b and k1, we concluded that the default values of 0.75 for b and 1.2 for k1 gives us the best recall rate for the corpus. A snapshot of the code snippet for the similarity class is shown below:

```
<schema name="bm" version="1.6">
  <uniqueKey>id</uniqueKey>
  <similarity class="solr.BM25SimilarityFactory">
    <str name="b">0.75</str>
    <str name="k1">1.2</str>
  </similarity>
```

For fetching the query results we have used for Extended DisMax Query Parser which processes phrases entered by users and searches each individual query terms in different fields based on the significance of each field. We have searched the query terms in two fields tweet_text and translated to provide multilingual searches to the user query.

Data Analysis and Visualization

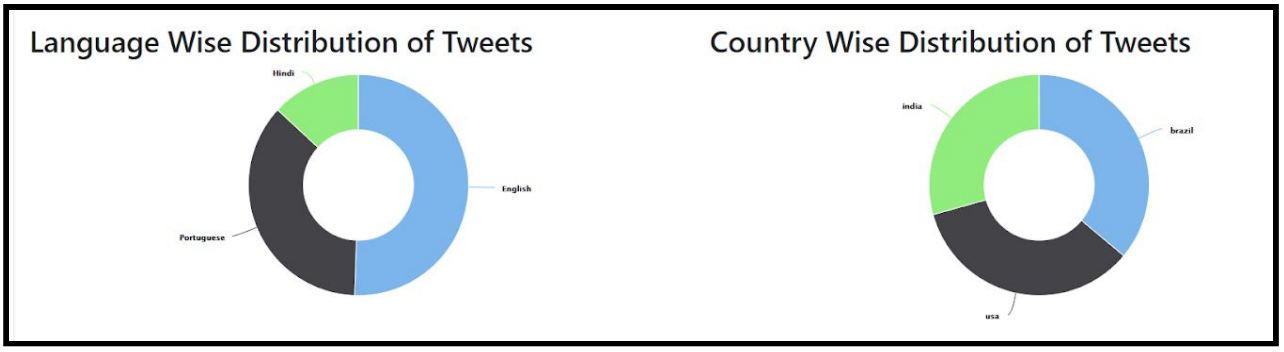
Based on user query we have used the facet feature of Solr to fetch the results for analytics. We have implemented nested faceting to fetch counts of one field based on another. This made the response time of the query very fast as instead of iterating over the tweets which just fetched the counts and analyzed the data.

The analytics are important to get a broader view of the search results and further information not easily perceived from plaintext query results. We implemented the following methodologies to analyze the collected data and display it in the form of time series, donut, pie and bar charts. We also performed sentiment analysis on tweets to generate the overall sentiment of each tweet. The graphs are rendered in real time based on the query. The data analysis methods we used are as follows:

Time series for POI tweets and the replies to the POI tweets:

Here the 3 lines indicate the 3 sentiments of the tweets. Green indicates positive, black indicates negative and blue indicates neutral.

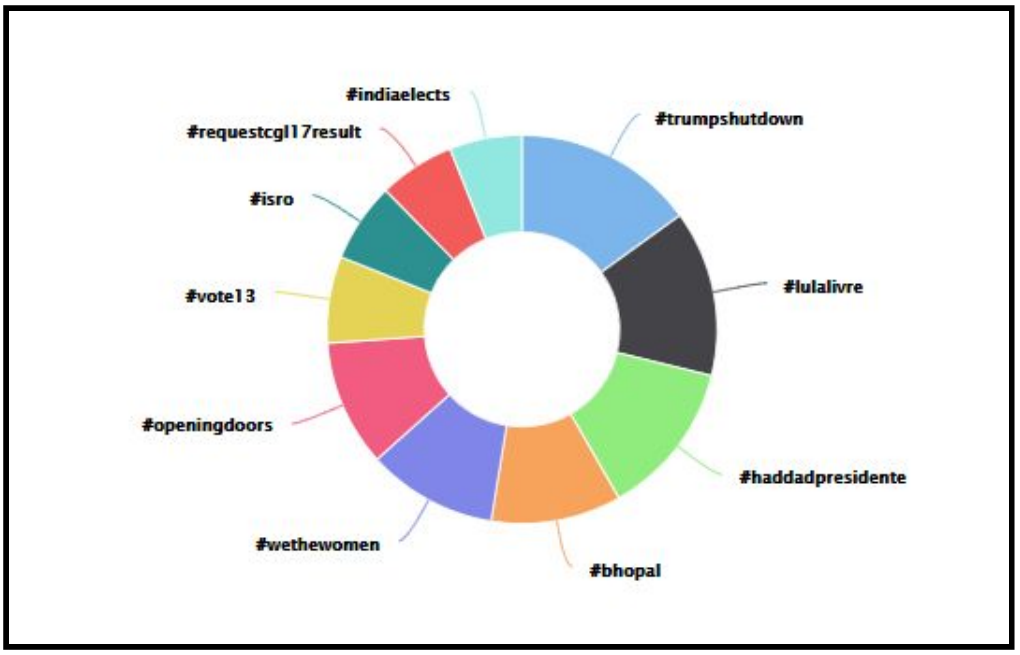
Donut chart for Language and Country wise distribution:



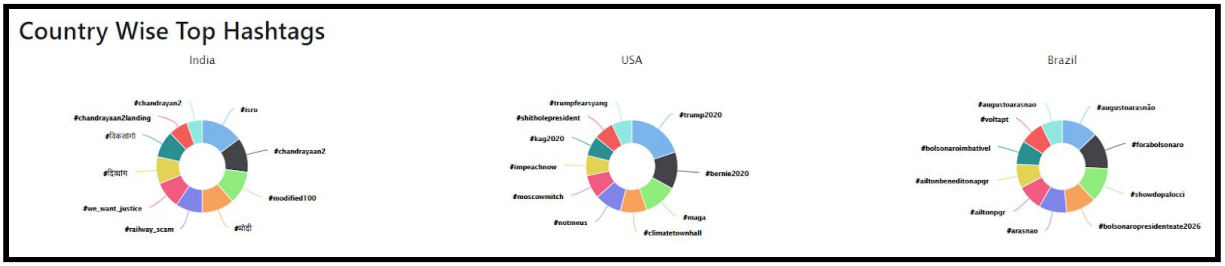
This chart gives the overall distribution of the language inwhich the query was tweeted and also the country wise distribution of tweets.

Donut Chart for Hashtags Distribution:

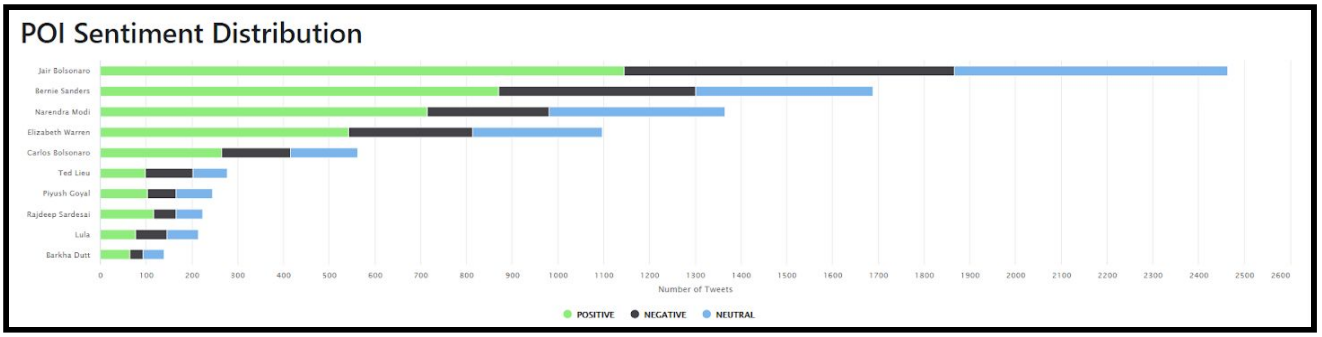
Distribution of hashtags are represented in donut charts. Sgown below is the distribution of top 10 hashtags for a query



Along with this, we have also represented the top hashtags from all 3 countries as shown below.If we search for one specific country then the overall hashtags for the other countries will be empty.

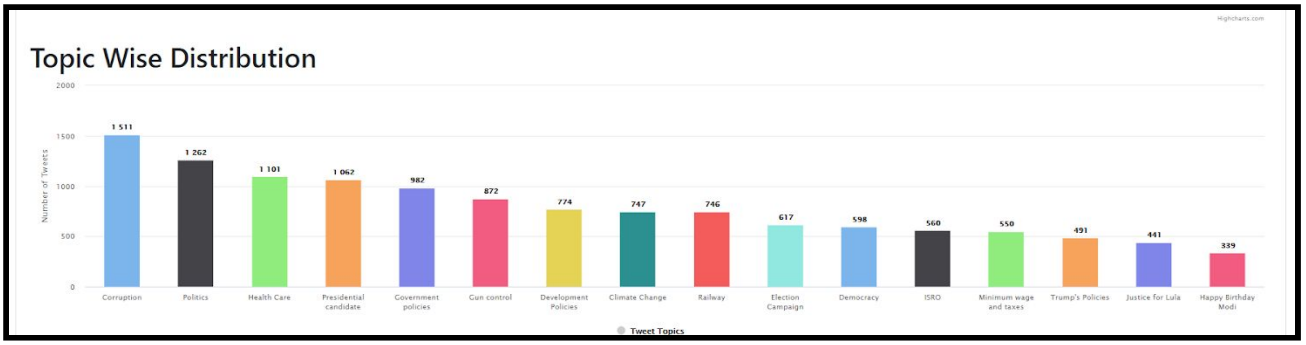


Stacked bar chart for POI Sentiment Distribution:



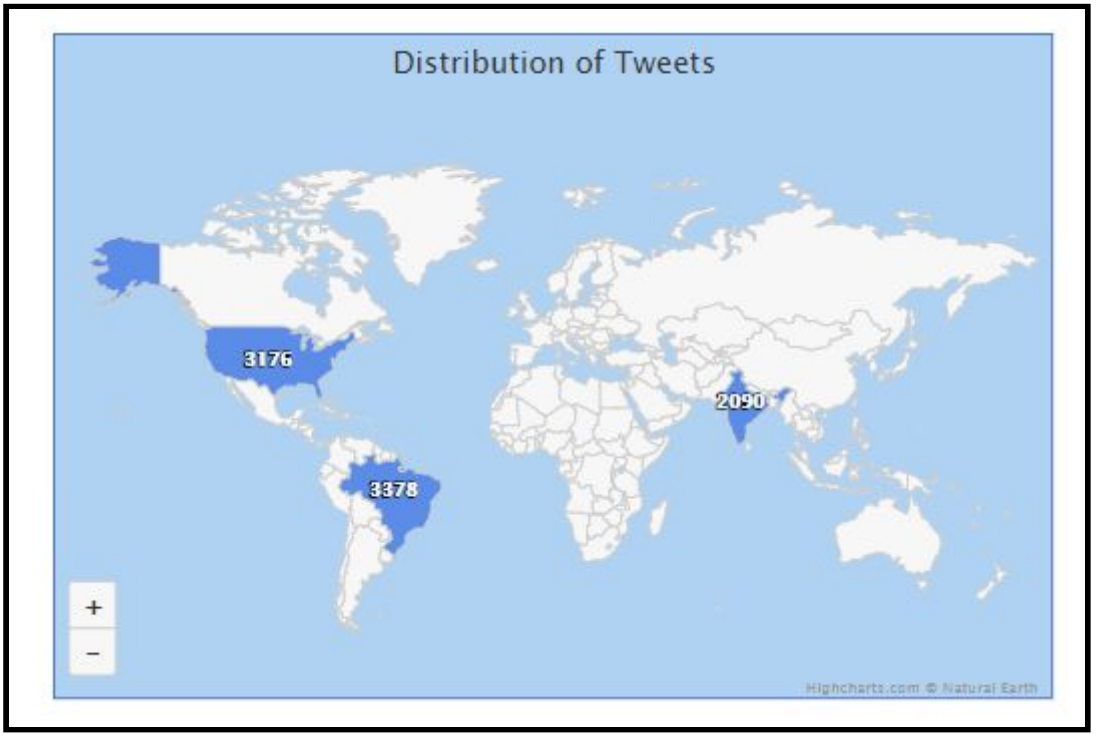
In this graph we have represented the POI sentiments for the tweets associated with the entered query. Each POI's tweet distribution for positive, negative, neutral sentiments are represented in green, black and blue color respectively.

Column Chart for Topic Wise Distribution:

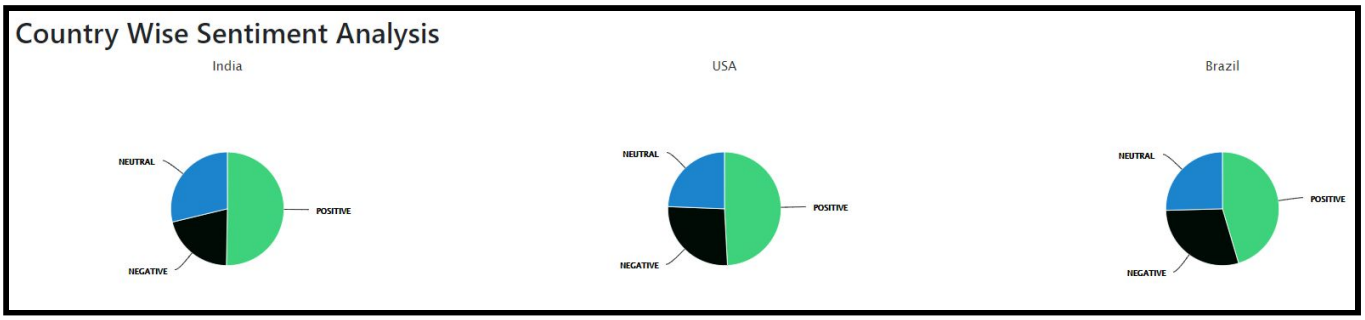


The graph shown above shows the topic wise distribution of tweets for the entered query. These topics are obtained from LDA and mapped to tweets. These topics can also be filtered from the filter tab to show only few selected topics.

Country Wise Distribution of Tweets:



The map above shows the total count of tweets in India, Brazil and the USA. This count included both the POI tweets and replies to POI tweets.



The above pie chart indicates the overall sentiments of tweets in all 3 countries. From the filter tab, we can choose to get results for any one specific sentiment as well.

UI Development

User interface and user experience are the most significant elements of any website in order to provide users with easy navigation across the website. It is the link between the user and the website.

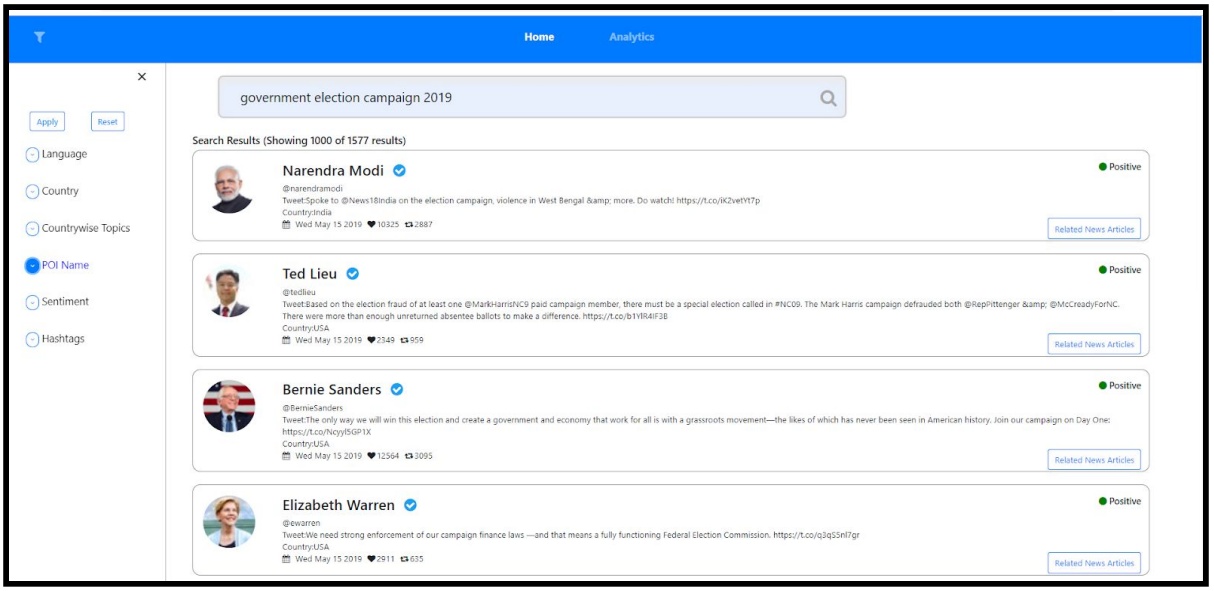
Our aim was to develop a simple and elegant user interface which makes its easier for the user to understand the various features our website has to offer. Keeping this in mind, we used Angular 6 web framework along with TypeScript and HTML.

Some of the features provided are:

1. Search box for the user to search tweets based on an input query
2. Home tab which displays the tweets as a result of the search query
3. Analytics tab which enables the user to visualize the data
4. Filters to narrow down the search results as per user's preferences.

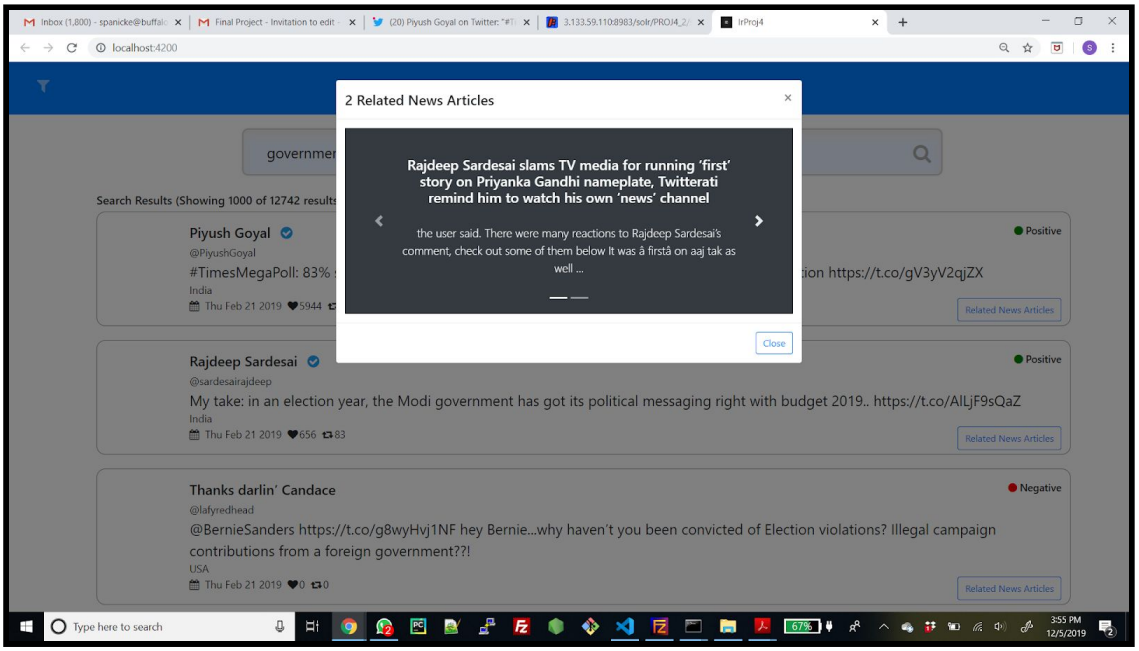
The below screenshot show the search results for the query 'government election campaign'.

The displayed results comes with the name of the person who tweeted, tweet date, tweet text, country, number of retweets and number of likes on that tweet.

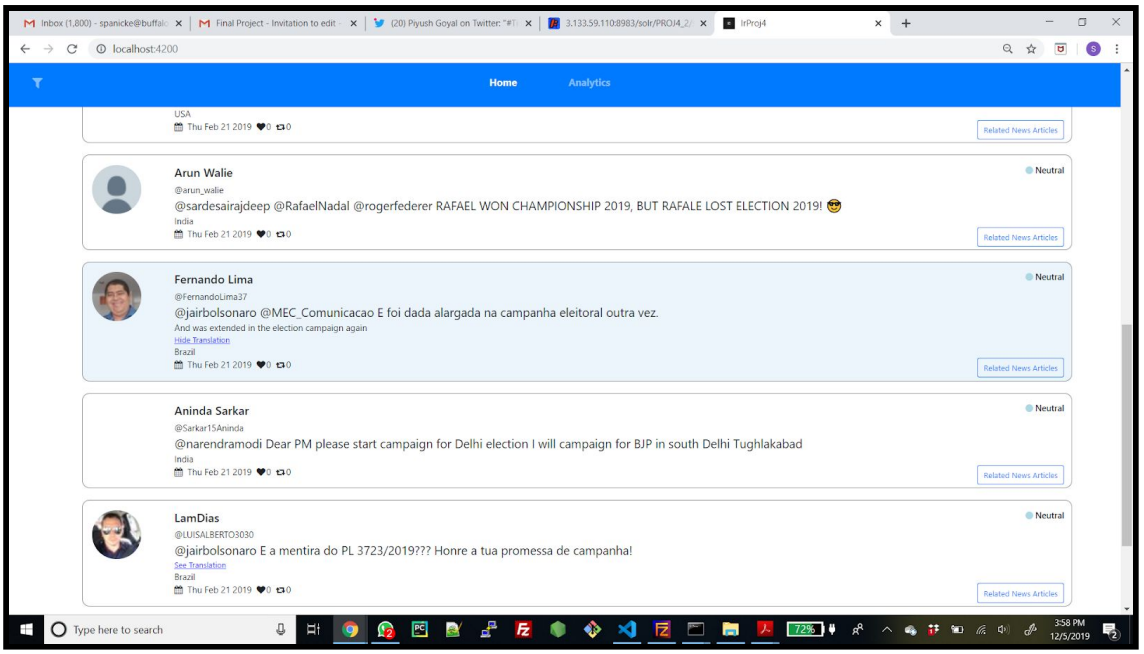


Each tweet result also has a button ‘Related News Articles’ which lets the user to have a look into the articles that are related to the person of interest associated with the tweet and published after that particular tweet date. We had fetched the articles based on the week after the tweet was posted.

However, after the feedback from faculty we are fetching the new articles for 1 month after the tweet was posted. This was done to get the impact of the person of interest’s tweet once it was posted.



Also, the user can see the translation of tweet text which is in a language other than English.



Video demonstration

The video demonstration for this project can be found at the link below:

https://www.youtube.com/watch?v=tV1nVDUjivY

Team contributions

Team Member	Contribution
Anushree Parmar	Solr Optimization and API's for fetching data from SOLR
Mansi Shetty	UI design and website functionality, News Articles
Mohd Ehtesham Shareef	Translation, sentiment analysis and topic modelling
Sneha Panicker	Data Analytics and Visualization

References

https://stackabuse.com/python-for-nlp-topic-modeling/

https://gnews.io/

https://www.highcharts.com/

https://angular.io/

http://flask.palletsprojects.com/en/1.1.x/

https://stackoverflow.com/

https://textblob.readthedocs.io/en/dev/

https://cloud.google.com/translate/

https://lucene.apache.org/solr/guide/

https://www.youtube.com/

https://lucene.apache.org/solr/guide/6_6/the-dismax-query-parser.html#the-dismax-query-parser