CSE535 Project 4: Analysis and Visualization of Twitter Data, HAMπ

Abstract

In recent times, Twitter has become the best place to voice individuals' opinions and it is observed that influential personalities are more active on Twitter these days. This has big impacts on our daily lives as reported in news media as well. Twitter has abundance of opinions and reactions about trending as well as debatable topics which can be stored and shared through APIs. Making use of the APIs, we index the fetched tweets using Solr for easy search and analysis of tweets and its replies. Analysing the public sentiment is used in many applications such as firms trying to find out the response of their products in the market, predicting political elections and predicting socioeconomic phenomena like stock exchange. The aim of this project is to use the sentiment analysis to predict the societal impact of tweets by politically influential people and crawl relevant and related news articles off the internet. We also segregate the tweet's replies as positive, negative & neutral for analysis purpose.

1. Introduction:

We aim to analyse the impact of political rhetoric globally and on social media. In our project we are extracting data from Twitter and performing analysis of the collected data based on the person of interest, country, language and time period. We have gathered around 200k tweets which mainly consists of tweets made by impactful people (persons of interest) each having at least 1000 tweets. These persons of interest belong to 3 different countries namely India, United States of America and Brazil. Our collected Twitter data is country specific and is mainly comprised of Hindi, English and Portuguese tweets. Our dataset also consists of replies to these tweets which we use to analyse the impact of a tweet by doing sentiment and volume analysis. We have displayed most discussed topics from our dataset by means of topic categorization. We have provided keyword search to the user to get tweets and perform keyword-based analysis. User can also perform a faceted search which is more specific and allows us to get filtered data, based on POI name, languages, date, etc.

1.1 Topic Analysis

Topic Models are a type of statistical language models used for uncovering hidden structure in a collection of texts. By doing topic modeling, we build clusters of words rather than clusters of texts. A text is thus a mixture of all the topics, each having a specific weight. Tagging, abstract "topics" that occur in a collection of documents that best represents the information in them. There are several existing algorithms you can use to perform the topic modeling. We are using Latent Dirichlet Allocation (LDA) model to extract important topics from our dataset. The below figure gives us a rough estimate of working of LDA model using which we retrieve main topics from our Twitter data collection.

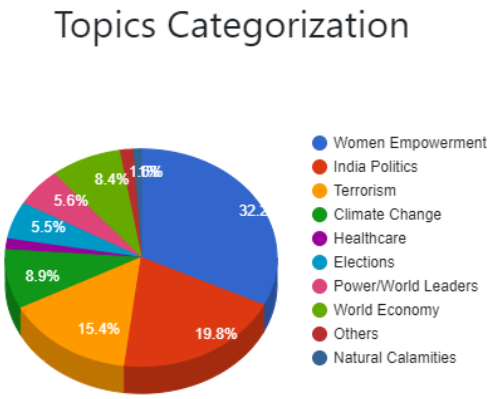

Figure 1: Topic Categorization

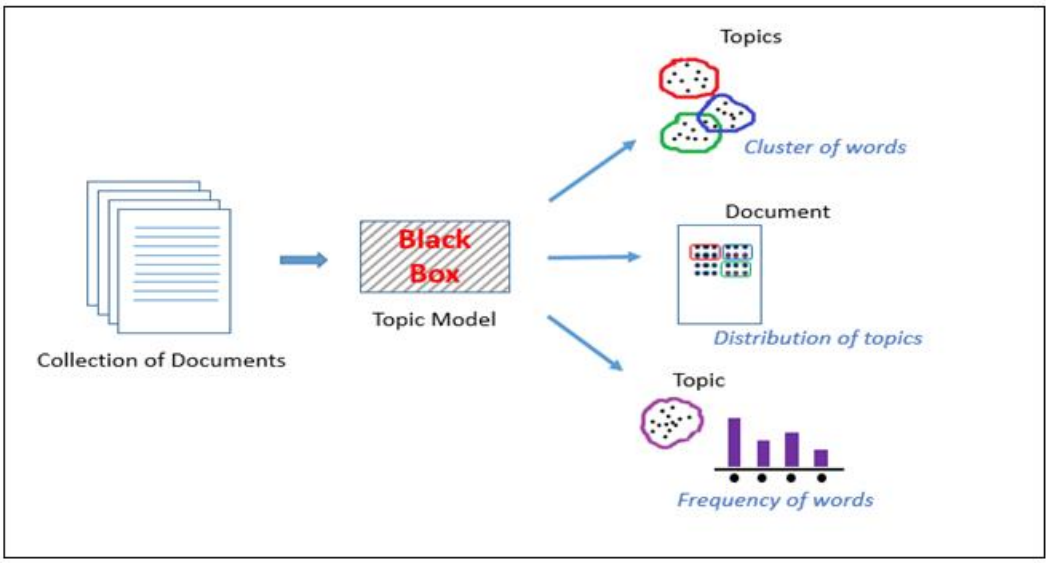

Figure 2: LDA based Topic Analysis

1.2 Searching Tweets

We have fetched tweets by 15+ influential political figures who have been very active on Twitter. And then fetched replies to these tweets. We indexed all these data in Solr using BM25 Similarity model and used various tokenizers and filters such as PorterStem. To fetch tweets of our persons of interest, we have created an UI which has a search bar which takes input as free text and have provided certain filters to the user. The filters include POIName, top 10 hashtags, From date, To date, etc. We have provided a checkbox to 'include Replies', on check of which we provide replies to the original tweets as well on our persons of interest.

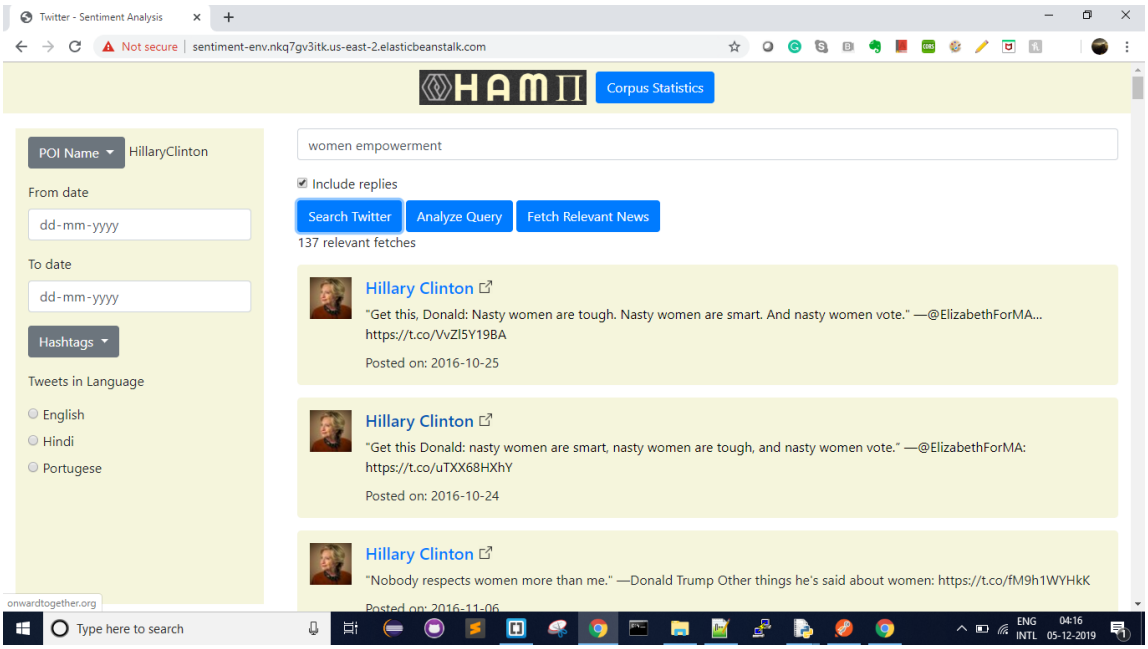

Figure 3: Tweet Search results

We also fetch respective metadata from Solr using facet fields to group field values of certain fields. From this metadata, we get top 10 hashtags, POI names and top 10 tweet dates on searched topic.

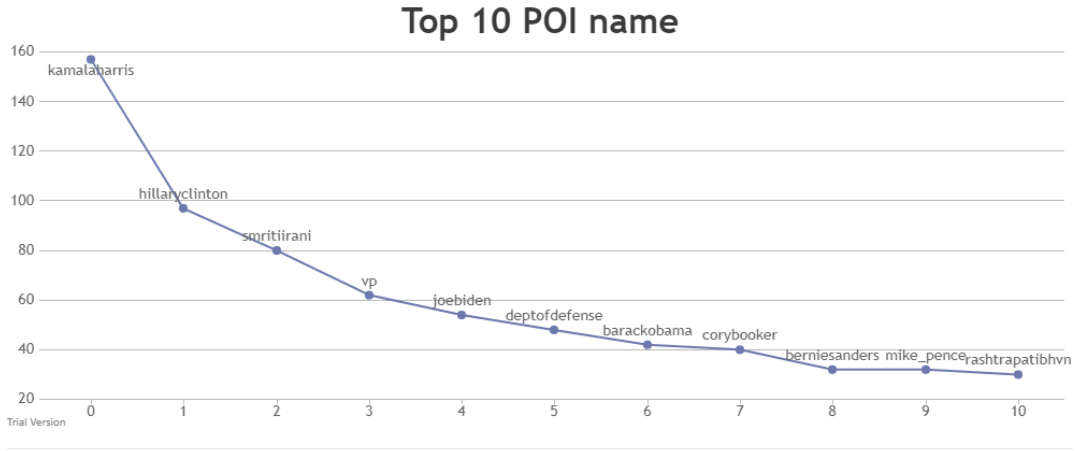

Figure 4: Top Fetch Results

1.3 Sentiment Analysis

We have processed original tweets posted on Twitter as well as replies to these tweets separately to understand the sentiments on the topic that user is interested in. We get the sentiment polarity in numbers with the help of a library called **TextBlob**. We have termed all the tweets with polarity values greater than 0 as positive sentiment, polarity value 0 as neutral sentiments and negative polarity values to depict negative sentiments. We calculate the sentiments for every search on search results obtained from Solr data and display in the form of a pie chart as shown below.

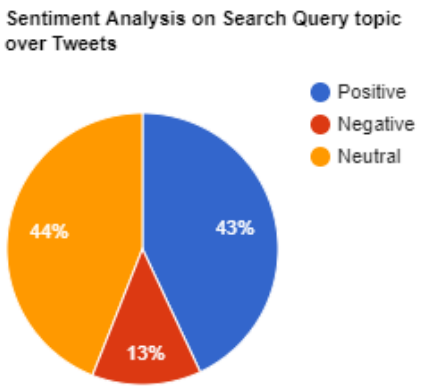

Figure 5: Sentiment Analysis pie chart

1.4 New Fetching based on Keyword Search

To find the impact created by the POI's tweets in the physical world, we crawl news articles around the days where maximum activity has been recorded on the particular tweet.

news query -> ('Twitter on ' + { search term } + { metadata[POIname] } + { metadata[hashtags] })

since -> metadata[tweet_date]

This gives us all the news articles involving news query posted after the tweet date on which maximum tweets are observed for the search term.

2. Flow Diagram

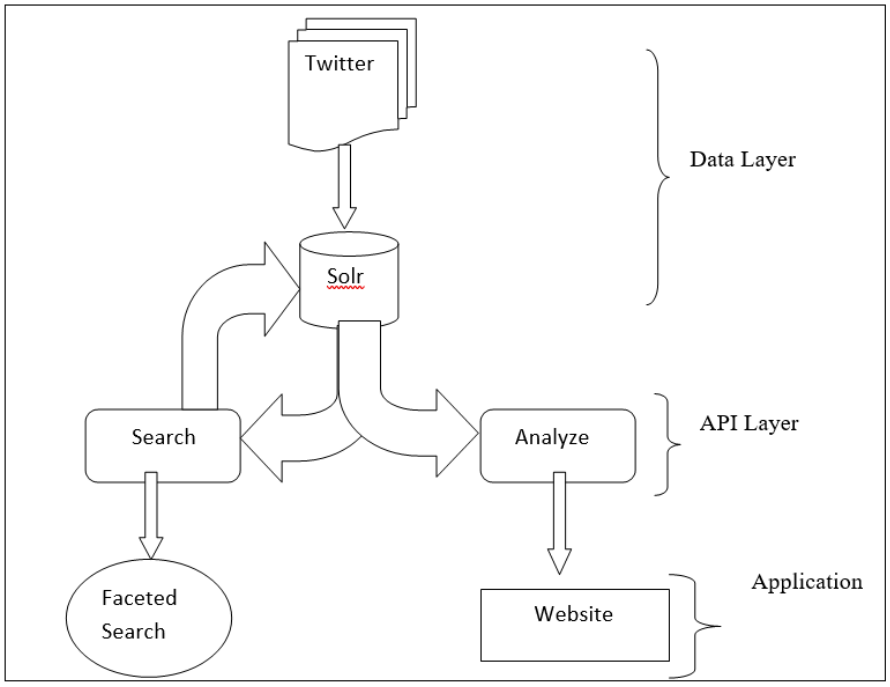

Figure 6: Data Flow Diagram

3. Technology Stack:

3.1 Backend Technologies Used:

- Apache Solr: Apache Solr is an open source search platform built upon a Java library called Lucene. We have used Solr in our project because it can index files and return desired results based on the user query.

- Python Flask: Flask is a lightweight web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It began as a simple wrapper around Werkzeug and Jinja and has become one of the most popular Python web application frameworks.

3.2 Front End Technologies Used:

We have used the following languages to design the UI of our website:

- HTML
- JavaScript
- Bootstrap
- Angular JS

3.3 Other APIs Used:

We have performed Sentiment Analysis using TextBlob:

TextBlob: TextBlob is a Python (2 and 3) library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.

We implemented Topic analysis using Latent Dirichlet Allocation (LDA) model:

LDA: Topic modelling is a type of statistical modelling for discovering the abstract "topics" that occur in a collection of documents. Latent Dirichlet Allocation (LDA) is an example of topic model and is used to classify text in a document to a particular topic. It builds a topic per document model and words per topic model, modelled as Dirichlet distributions.

4. Results and Reflections:

4.1 Landing page of website

http://sentiment-env.nkq7gv3itk.us-east-2.elasticbeanstalk.com/

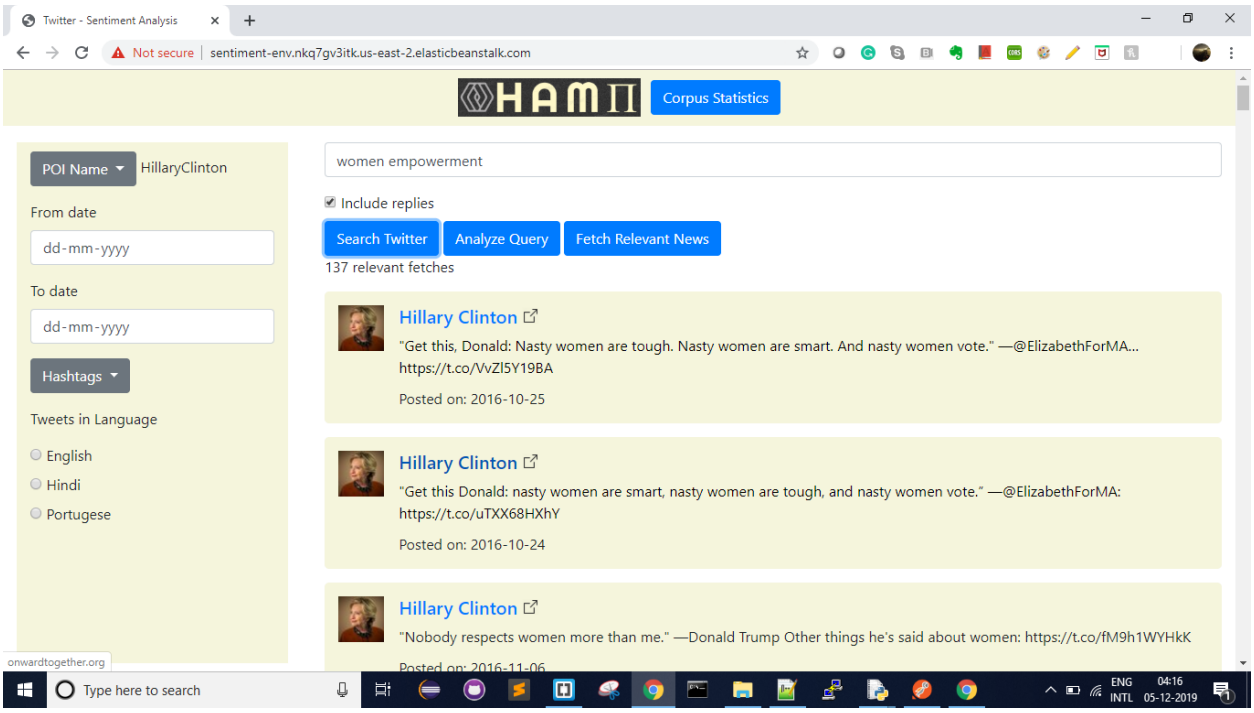

Figure 7: Landing Page

5. Group Contribution

Task	Sub-Tasks	Contributor(s)
Dataset & Understanding Solr		Astrid Gomes
AWS		Harshali Talele
Front End:		
	Visualization & Analytics	Astrid Gomes
	Keywod & Faceted Search	Preeti Kumari
	Relevant News Articles	Mansi Wagh
Topic Analysis		Preeti Kumari
Back End:		
	News Scrapping	Harshali Talele
	Sentiment Analysis	Mansi Wagh
Demo Video		Mansi & Preeti
Report		Harshali & Astrid

6. Conclusion

HAMπ Twitter Analysis Search allows a user to search a particular query on our corpus. We have implemented faceted and query search using Solr. We present sentiment analysis on the query and various line graphs to represent top hashtags, maximum tweet days and POIs with higher tweets on the search query. The analysis on the corpus gives topic categorization, percentage of tweets from the 3 countries with lingual distribution as well as the top 10 hashtags in the corpus. We give relevant news fetches using twitter and POI name as keywords for crawling news on the internet.

7. References

[1] https://lucene.apache.org/solr/6_3_0/index.html

[2] https://textblob.readthedocs.io/en/dev/

[3] https://towardsdatascience.com/topic-modeling-and-latent-dirichlet-allocation-in-python-9bf156893c24

[4] https://www.fullstackpython.com/flask.html

[5] https://docs.aws.amazon.com/ec2/index.html?nc2=h_ql_doc_ec2

8. Contributors

Astrid Gomes	50317492	astridyv@buffalo.edu
Harshali Talele	50318248	harshali@buffalo.edu
Mansi Wagh	50318704	mansiwag@buffalo.edu
Preeti Kumari	50321294	preetiku@buffalo.edu