Sentiment Analysis on Twitter Data

FIT5147 - Data Exploration and Visualisation

Visualisation Assignment

Case Study:

Elections in India (Lokh-Sabha Elections - 2019)

Roopak Thiyyathuparambil Jayachandran - 28 April 2019
Student Id : 29567467
Tutor : Joy Zhao

Introduction

Democracy is a participative political system, through which a country is governed by representatives validated by popular votes. Even though political leaders can be held accountable for their activities in retrospection, citizens active participation is limited to the voting stage in a democratic setup. Hence, comprehensive analysis of voting patterns and practices would form a crucial step in democratic practices.

Lok Sabha election is one such democratic exercise which takes place periodically over a period of five years in India. Being the world's biggest democracy in terms of population and sheer size, detailed analysis and research on voting patterns would form but a rudimentary exercise in ensuring transmission of citizens' voice.

Lok Sabha elections are conducted every five years to elect the lower house members known as Members of Parliament (MP). Candidates contest in elections from their respective constituencies and every adult citizen can cast vote for their constituency. The 17th Lok Sabha elections were conducted in India from 11 April 2019 to 19 May 2019. This Visualisation Project attempts to identify the sentiments of Twitter users of major parties and check how these sentiments are related to different regions and keywords, thereby capturing a bird's eye view on the Indian election system. The insights generated would lay a strong foundation in breaking down the extremely complex system of Indian election system for further analyses.

Identified Audience

With this Visualisation Analysis, it is intended to identify the online popularity of one political party over the other. Twitter Sentiment Analysis is used as a tool for this.

It will be helpful for the party stakeholders as well as general audience on different verticals.

We have identified different aspects for comparison during the exploration stage of the project some of which are :

- Region wise popularity of a party
- Key terms which a party focuses on.
- Overall sentiment comparison
- Online impression comparison (Retweets, likes)

These comparisons can help the stakeholders to identify what audience have liked and plan their future campaigning accordingly targeting on their weak points and strengthening the strong areas.

Extraction Python Code

```python
import tweepy
import re
from textblob import TextBlob
import pandas as pd
from wordcloud import WordCloud
from nltk.tokenize import RegexpTokenizer
from matplotlib import pyplot as plt

consKey = "4TGPfg62pT76yWTnTE8INjJCL"
consSecret = "dkhHybaFTIAflvCVB5vFHQPu0eCHFQN1BIbpdx3hjt8q3aRqWy"
accessKey = "3061686882-PrrngsYGKlMV92KseseATs1Gjr5Yyucry61Lwl2"
accessSecret = "aLOiD62JKQg29mSoz6chfpLP7fbBpG4syMuaUvlAqe9J0"

auth = tweepy.OAuthHandler(consumer_key=consKey, consumer_secret=consSecret)
auth.set_access_token(_accessKey,  accessSecret)
api = tweepy.API(auth, wait_on_rate_limit=True)
tweets = tweepy.Cursor(api.search, q="BJP", lang="en").items(1000)

api = []
text = []
created_date = []
lati = []
longi = []
polarity = []
impressions = []
pol_value = []
followers = []
user_location = []

def extractCoordinates(stri):
    a = re.findall(r"coordinates=.+?\)", stri, re.DOTALL)
    if len(a) > 0:
        b = re.findall(r"\[.*?\]",a[0], re.DOTALL)
        if len(b) > 0:
            c = b[0][3:-1]

        d = c.split(",")
        lat = d[0]
        lon = d[1]
        return lon, lat
    else:
        return None,None
```

```python
count = 0

for each in tweets:

    api.append(each)
    if each.place == None:
        continue
    count += 1
    created_date.append(each.created_at)
    text.append(each.text)
    longi.append(extractCoordinates(str(each.place))[0])
    lati.append(extractCoordinates(str(each.place))[1])
    impressions.append(each.retweet_count + each.user.favourites_count)
    followers.append(each.user.followers_count)
    user_location.append(each.user.location)


    emotion = TextBlob(each.text)
    emo = emotion.sentiment.polarity
    if emo >= 0.5:
        polarity.append("Highly Positive")
    if emo <= -0.5:
        polarity.append("Highly Negative")
    if emo == 0:
        polarity.append("Neutral")
    if 0 < emo < 0.5:
        polarity.append("Positive")
    if -0.5 < emo < 0:
        polarity.append("Negative")

    pol_value.append(round(emo, 2))

BJPDict = {'text': text, 'created_date':created_date, 'latitude':lati, 'longitude':longi, 'retweet':impressions,
                    'polarity':polarity, 'pol_value': pol_value, 'followers': followers, 'location':user_location}

BJP = pd.DataFrame(BJPDict)
print(BJP.head())


BJP.to_csv("bjp_loc3.csv", sep=',', encoding='utf-8')
```

Design - Five Sheet Design Methodology

Five sheet design methodology is used to create the design for this Visualisation task. This technique helps in planning and identifying different alternatives which was later filtered and combined to increase the effectiveness. Basically comprises of 3 stages of design creation :

- Brain storming (Page 1)
- Design Alternatives (Page 2, 3 , 4)
- Realisation (Page 5)

Page 1:

In the first page all the possible visualisation techniques have been identified and drawn using pencil and paper.

To visualise each of the scenario mentioned in the previous page we have identified some of the plots as a part of brain storming. These are Bubble chart, Line chart, India map, India map Choropleth, Bar chart, Density plots Pie chart, Word cloud, stacked bar chart etc. Each of the plot can be used to visualise information but need to filter only the plots which can display the findings effectively. Keeping in mind the verticals i.e,

- Region wise popularity of a party
- Key terms which a party focuses on.
- Overall sentiment comparison
- Online impression comparison (Retweets, likes)

Some of the plots have been filtered. Some are also combined to show the visualisation effectively. Like Line chart and calendar, etc

Page 2:

Each of the page 2, 3 and 4 of Five sheet design are divided into Subsections sections mainly big picture, Components and Discussion.

Components/Features :

- Drop down to choose between BJP and Congress
- A button to choose the visualisation
- Line graph to show day wise trend
- Chloropeth map to show location wise analysis
- Bubble chart for online impression
- Density plot for each sentiment in a single plot

Discussion:

Positives: For each party a complete data visualisation will be displayed

Negatives:

- Difficult for comparison
- Density plot for each sentiment will show too much data

Considering these negatives, alternative design in page 3 has been created

Page 3:
As seen in the Appendix Page3 big picture has an option to select between BJP and Congress. On pressing the go button, on basic comparison visualisation is plotted. There will be also an option to see in detail for each party.

Components/Features :
- Multi-page web application
- Basic comparison on first page and detailed visualisation in the next.
- Single density plot for polarity calculated.

Discussion:
Positives:
- Basic comparison has been added
- Replacing pie chart with bar chart gives more clarity for similar values
- Single density plot more clear to understand.

Negatives:
- Complete visualisation in all grounds still not available.

Considering these negatives, and summing up the positives in page 2 and 3 alternative design in page 4 has been created.

Page 4:
Components/Features :
- Single page direct comparison representation
- India map with label used instead of Choloropeth Map

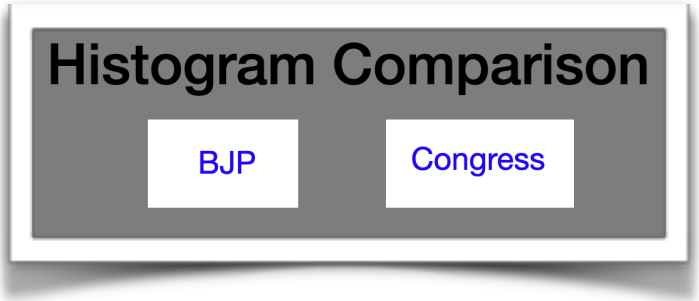
Discussion:
Positives:
- Complete comparison available
- World map shows better data

Negatives:
- The basic design is less interactive for the audience as all the comparisons and visualisations are shown from the beginning .
- Too much of data can add confusions while interpreting.

Page 5:
Page 5 is the realisation page which is the final design for implementation. This page is created combining the positives in all the alternative design sheets and filtering out the negatives.

Components/Features :

- Removed the date section as data was very limited for each date. So combined all data to get a bigger data.
- Single page visualisation having 5 comparisons:
- Histogram : To display how many tweets for each polarity. Polarity classifier into 5 buckets mainly Positive, Negative, Neutral, Highly Positive and Highly Negative
- India map : To plot the coordinates with the sentiment. This visualisation was done with a limited data as only a few tweets have coordinate information attached due to privacy concerns by Twitter Users
- Word Cloud : To display the key terms used by each party for campaigning which was identified by the twitter users.
- Bubble Chart : To display which sentiment has more online impressions(retweets, likes etc)
- Density Plot : To display the density of Polarity
- For each of the visualisation there is a selection option to select between BJP or CongressLike fig



Implementation

Data Extraction :

The main source of the data is Twitter. Twitter provides API for developers which can be used to extract data in Json Format. For this Exploration project **Tweepy** package in python is used to extract the data. Extracted data is then filtered to obtain just the required fields and other fields for explorations are generated through manual and natural language processing techniques.
In order to extract data from twitter, application has to be created in the official site which will generate consumer key, consumer secret key, access key, access secret key. These keys are needed which extracting the data. OAuthHandler and Cursor function from Tweepy are used with appropriate parameters. In the Cursor function the "keywords" and "number of tweets " is also specified. Twitter provides limitations to the extraction of large number of data for free but otherwise it is free of cost.

Extracted data is in Json format. Required fields are identified and data wrangling is done to convert semi structured data into structured data format.

Visualisation:

Python, R and D3 is used to create the visualisation through a web application.

Python used to create word cloud.
Reason:
- From the extracted data, the tweets were added up to make vocabulary and duplicates were removed. NLTK package was then used to remove stop words. Most frequent words were extracted and then word cloud was made using "WordCloud" python package. These images were loaded in the web application.

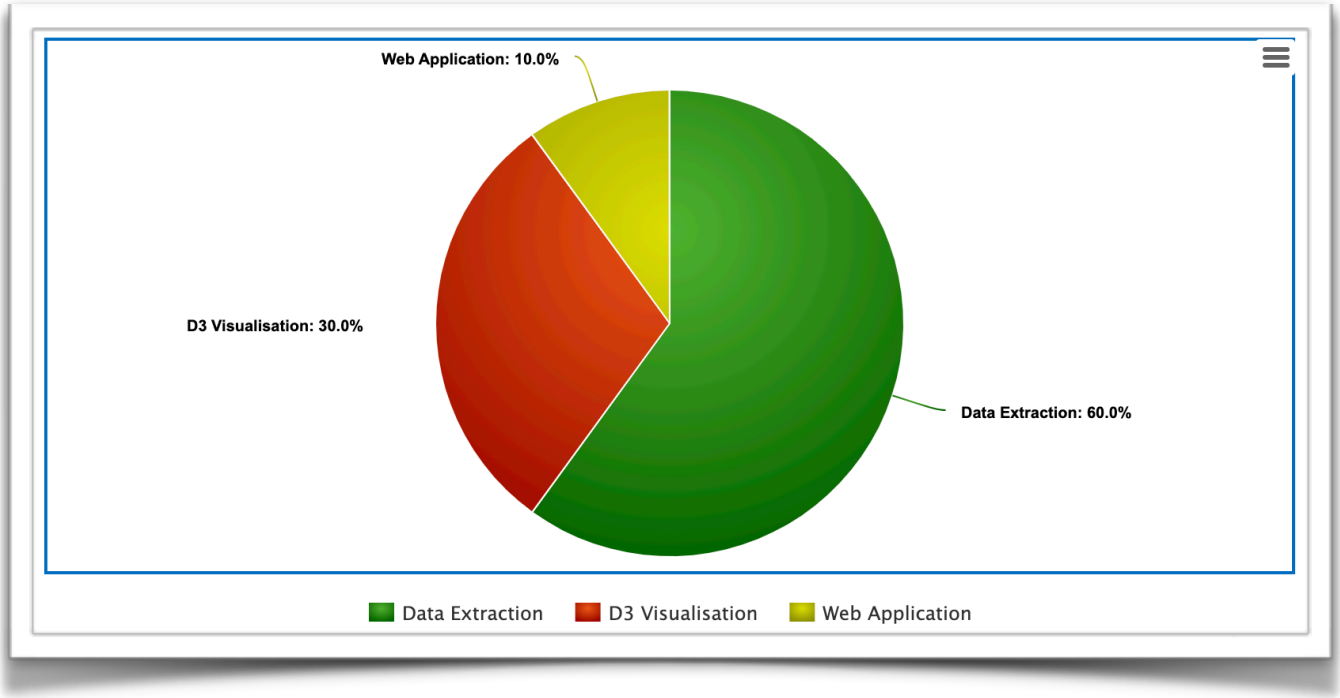
Reason to choose D3:
- D3 is good choice for web application visualisation
- Have more options to make the visualisation more interactive
- Online support for D3 package is good
- Lots of problems discussed in StackOverFlow and other platforms.

For web application:
Basic HTML with bootstrap for styling.

Data source:
Extracted tweets data source : "Cong.csv" and "BJP.csv"
D3 SVGs used:
- Each of the plots are plotted in a separate SVG blocks.

- Rect, Circle, Text are mainly used for implementation.
- On top of each plots, section boxes are added to make the plots more interactive.

Degree of Difficulty:



Data Extraction was the most difficult part of the project. Some of the difficulties during extraction are:

- Twitter setting limitation on extraction using API. 2000 tweets a day.
- Most of the data had user coordinates missing due to privacy settings
- Twitter data API returns data in JSON format. Extraction of relevant information was a bit tricky
- Sentiment Analysis task using TextBlob and NLTK python packages and finding the polarity for each tweet.

D3 difficulties:

- Page resize issues.
- Version problems. Some function names have been changed in each of the version which caused some difficulty in understanding. For example scaleLinear() and scale.linear() both are used for same purpose in different versions of D3.
- Adding for interactions were tricky.

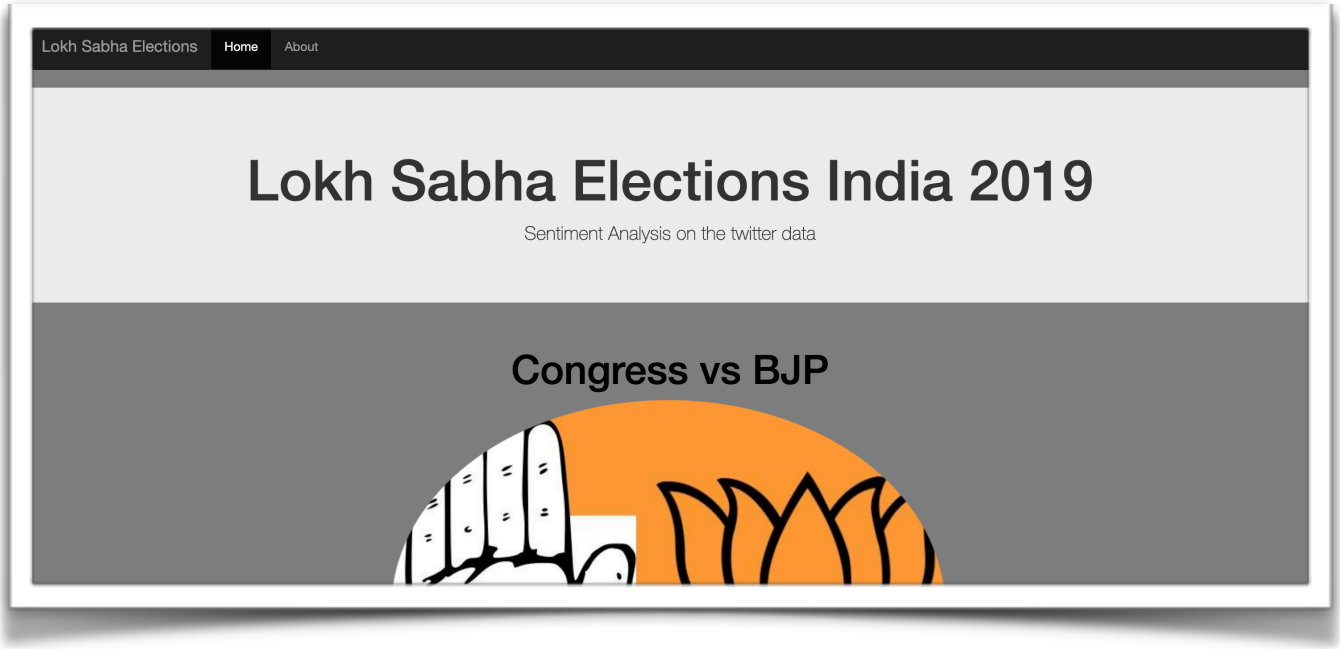
Web application difficulties:

- Proper lay outing and styling. Bootstrap was used as external css to style the web page.

User Guide

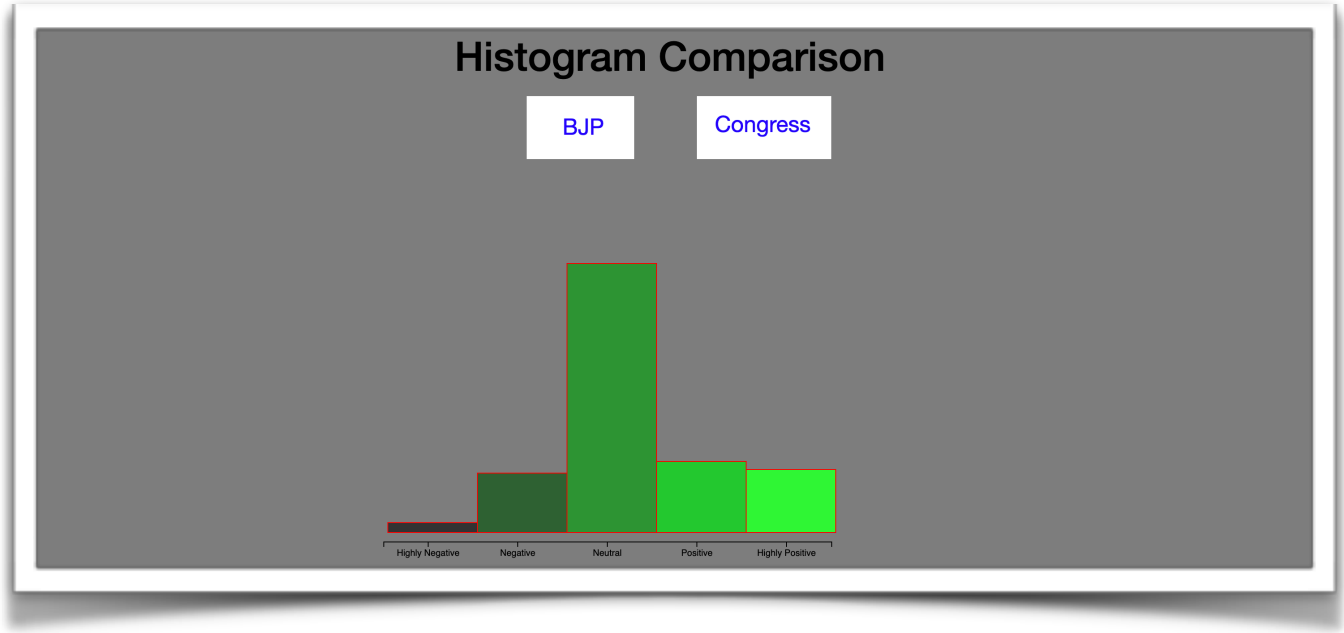
- Web page is a single page application which can be scrolled down to view all the content.

Home page

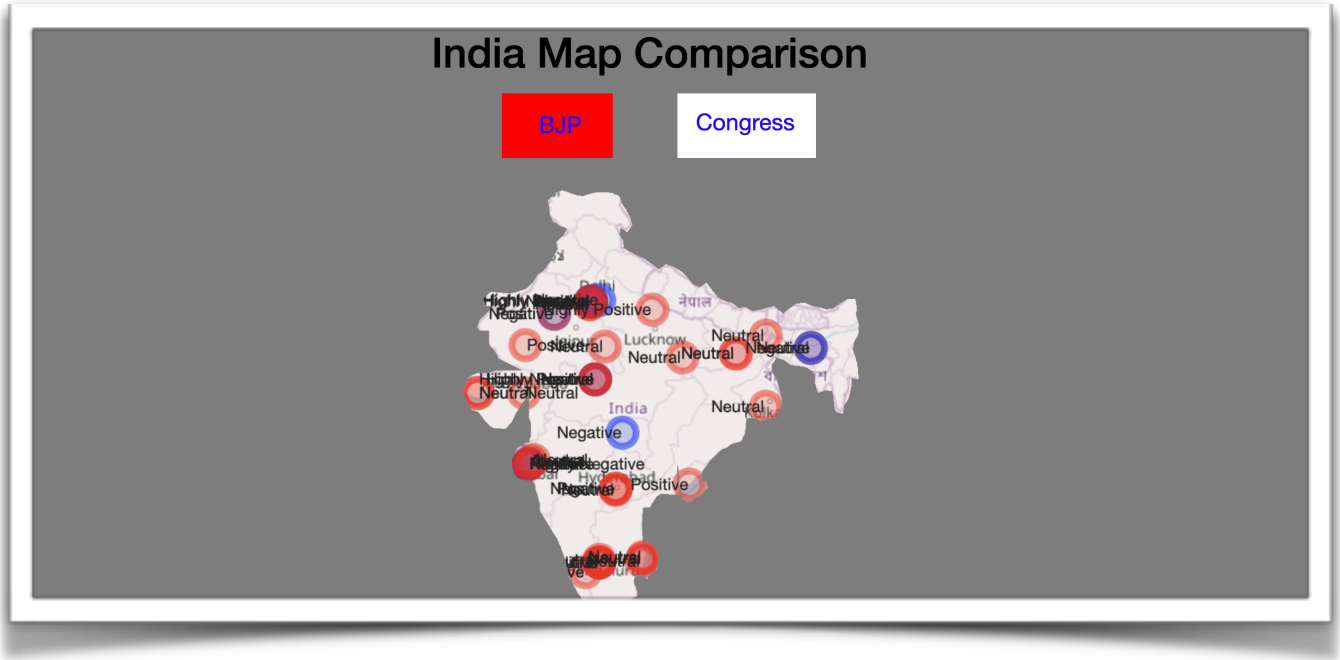


Initial view has a top menu which has 2 options: Home and About. About page will redirect to the authors portfolio page.

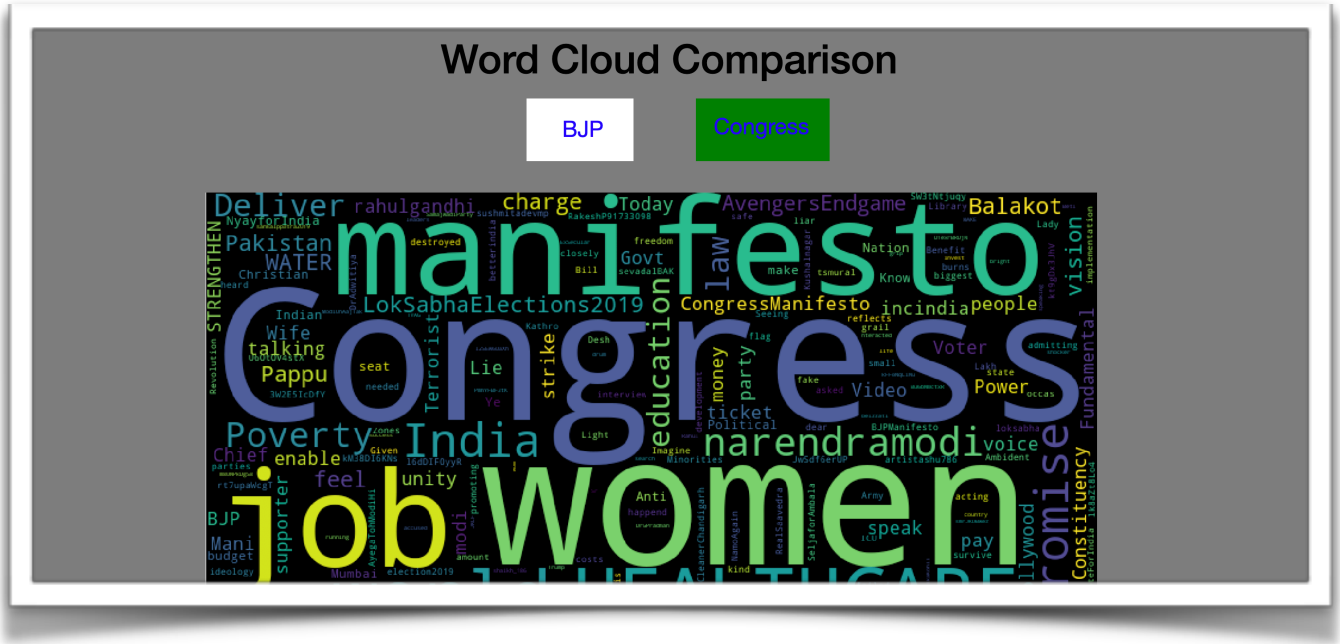
First comparison is a histogram of sentiments. Audience have an option to select histograms of their choice. A point to be noted here is that once BJP or Congress is selected, all visualisation will display its particular plot of that party. Since we want an overall picture of the scenario, frequency axis which shows the number of tweets is not relevant.



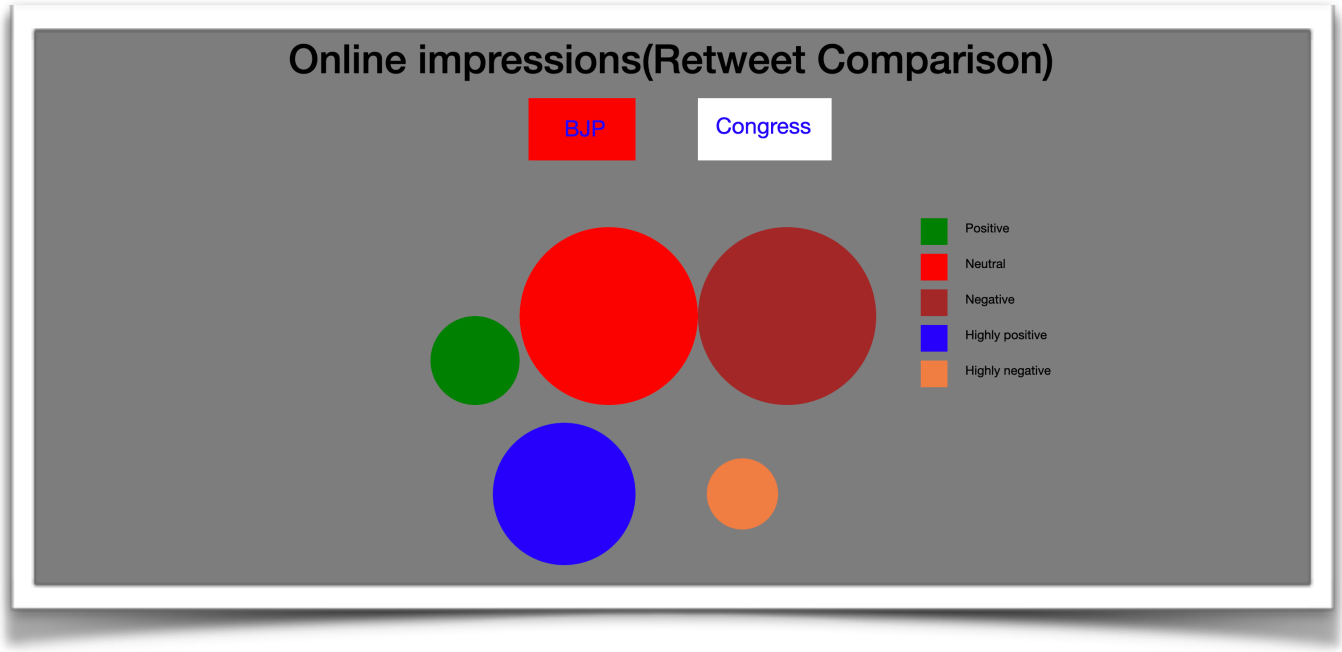
This is followed by India map comparison which visualises on which area people are tweeting for or against any party. Here we have option to choose between 2 to see the comparison



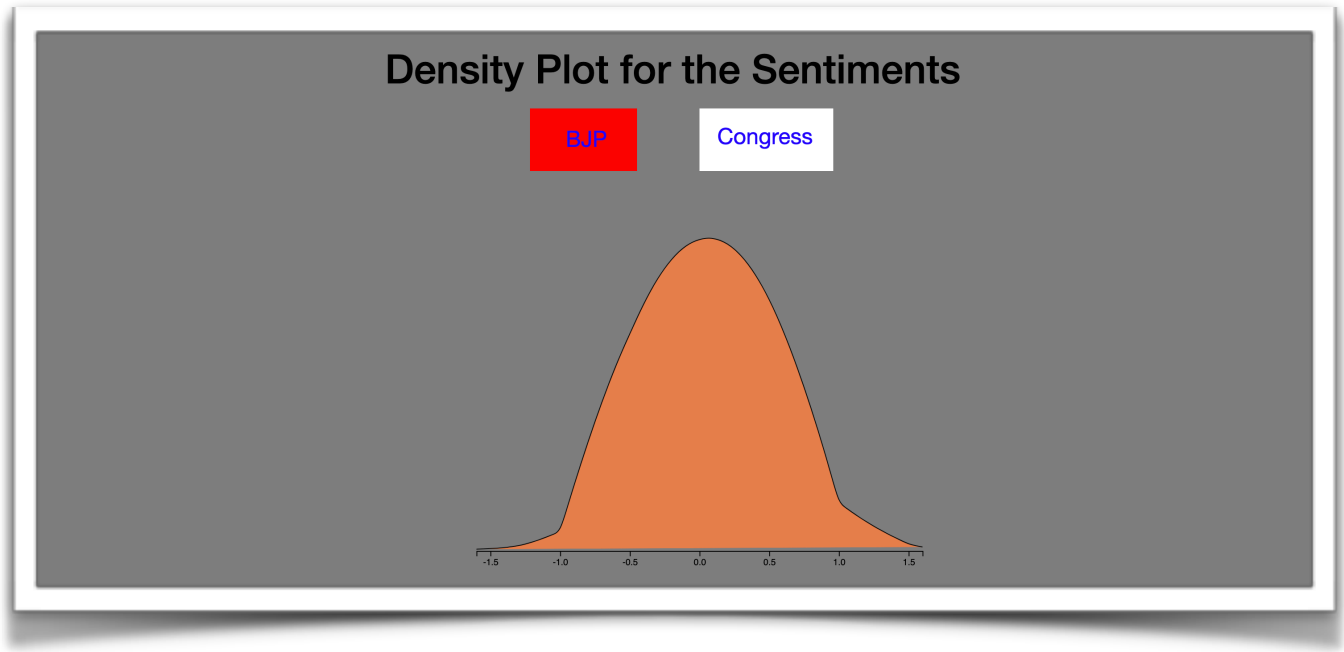
Area wise visualisation is followed by a word cloud which is generated using Python and imported to D3 as image. Here on selecting each of the party, their corresponding word cloud is visualised.



Word cloud is followed by Bubble chart of online impression which is a representation of retweets, likes etc for each sentiment. Here on selecting each of the party, their corresponding Bubble chart is visualised. A legend has also been created using rect svg.



Final visualisation is a density plot where the distribution of polarity for each party can be seen.



Conclusion and Limitation

1000 tweets pertaining to each political parties were the chosen sample size for performing the Visualisation analysis. It is acknowledged that such a number would be insufficient to derive meaningful insights from the Visualisation. Overall, results from the analysis would not find parity to realistic/eventual projections. Moreover, the demographic decomposition of eligible voters who express their political opinions on a specific social medium such as twitter would prove to be insufficient to derive meaningful insights.

It is identified that sentiment analysis and resulting Visualisations would help the targeted audiences gain a peripheral understanding of citizens' general political perception. However this would be limited to opinions of levels of satisfaction and comfort with the incumbent government and vague comparisons of major political parties. As these cannot be forged into metrics that drive meaningful actions, the validity of such an exercise would be limited to gaining an overall idea about the political scenario.

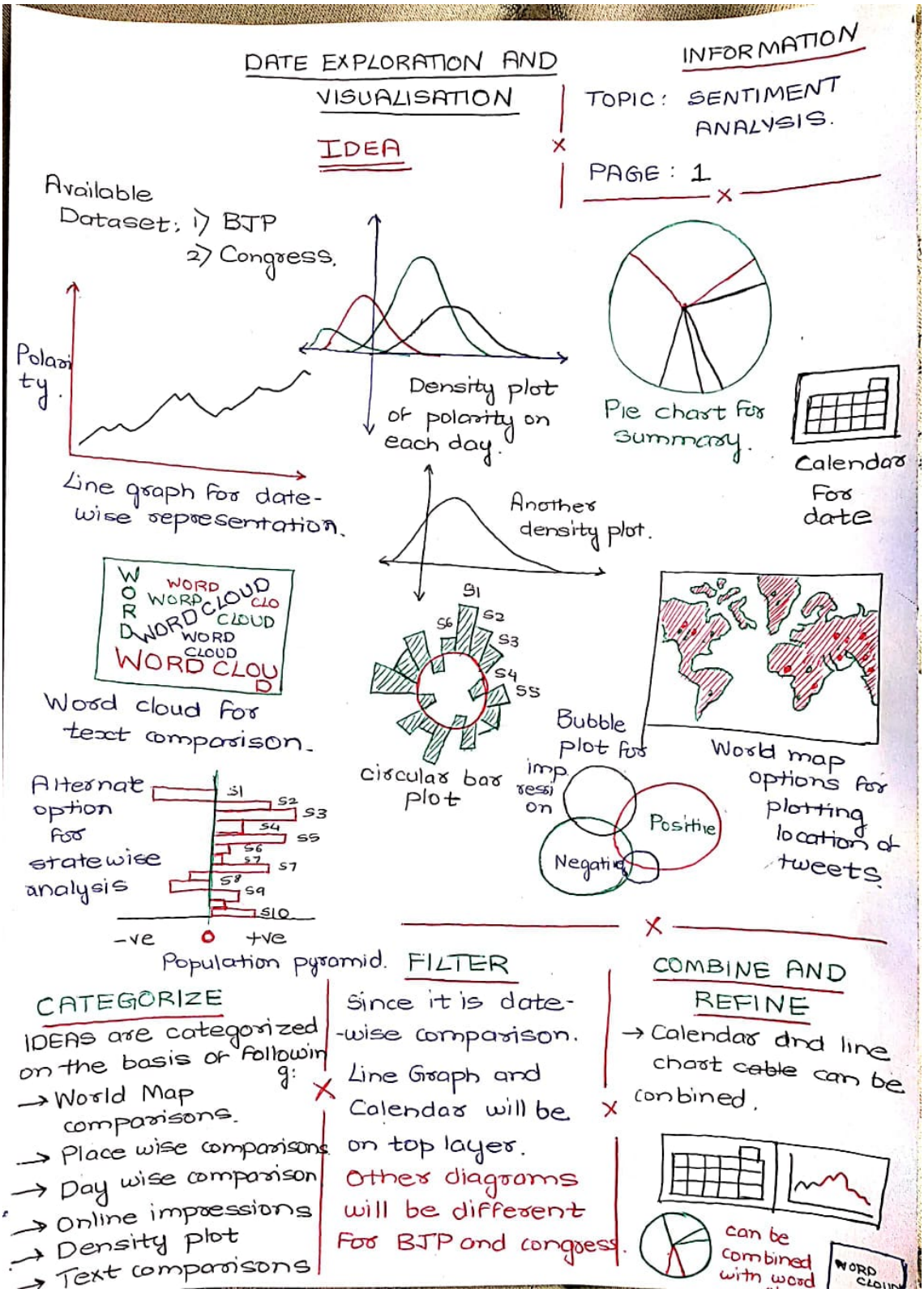
For accurate prediction of elections, dedicated teams have to be employed for real time analytics. Resulting reports could thus by utilised by parties and their Public Relations firms during the elections to propel them ahead.

References

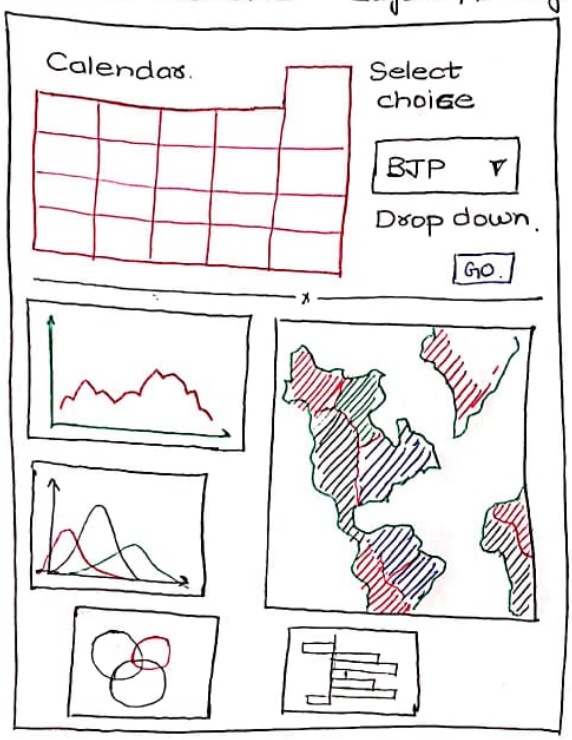
[1] TJ, R. (2019). Docs. Retrieved from https://developer.twitter.com/en/docs.html
[2] TJ, R. (2019). The R Graph Gallery. Retrieved from https://www.r-graph-gallery.com/
[3]TJ, R. (2019). d3/d3. [online] GitHub. Available at: https://github.com/d3/d3/wiki/Gallery.

Appendix

FDS - Page 1



BIG PICTURE Layout / Design

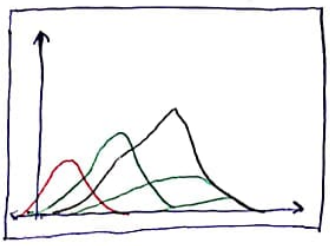


INFORMATION

DATE: 29th MAY, 2019
SHEET: 2
TOPIC: SENTIMENT ANALYSIS.

COMPONENTS.

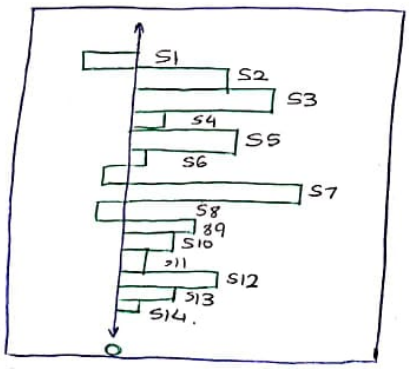
→ A calendar to select the date.
→ Drop down to choose between BJP and Congress.
→ A Button to display the visualization.
→ Line graph to show day wise trend.
→ Chloropleth map to show location wise analysis.
→ Density plot for each emotion.
→ Bubble chart for impression.



Density plot showing density of each emotion/ sentiment

COMPO-NENTS.

State wise average sentiment.



DISCUSSION

POSITIVES:-

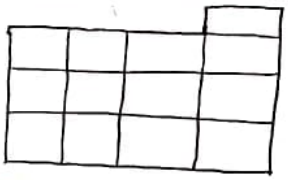
→ For each party, complete data w/ representation will be shown.

NEGATIVES:-

→ Difficult for comparison.
→ Density plot for each sentiment has too much data.
→ Chloropeth map shows no data on coordinates.

BIG PICTURE

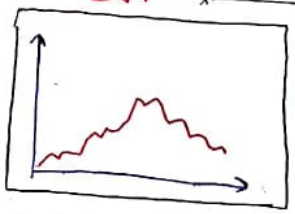
LAYOUT.

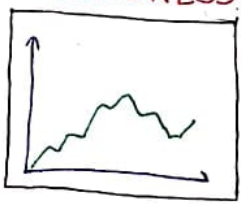


GO

BJP

CONGRESS



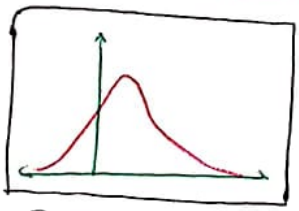


DETAIL

DETAIL

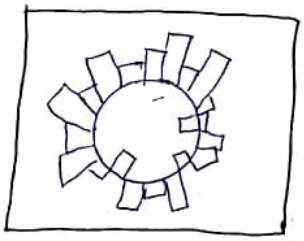
NEXT PAGE

BJP



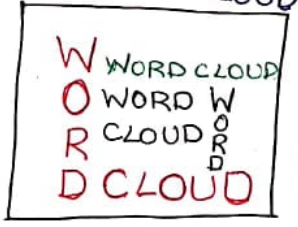
Density plot.

Polarity bar chart.



State

WORD CLOUD



INFORMATION

DATE : 29th MAY, 2019

SHEET : 43

TOPIC : SENTIMENT ANALYSIS.

COMPONENTS.

1) Multi-paged application.

2) Basic comparison on single page. Detailed visualisation on the other page.

3) Single density plot, instead of plot for each polarity used.

DISCUSSION

POSITIVE:-

1) Basic comparison added.

2) Replacing pie chart with bar-chart gives more clarity.

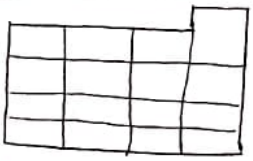
NEGATIVE:

1) Complete visualisation of comparison is not available.

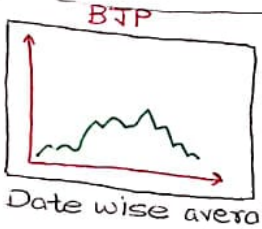
2)

BIG PICTURE Layout / Design

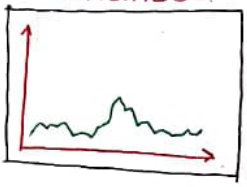
Select Date



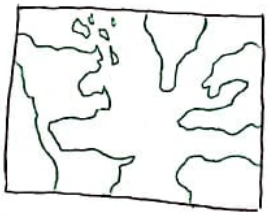
BJP



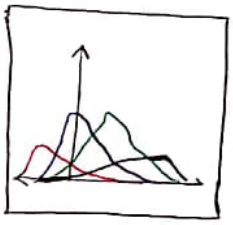
CONGRESS

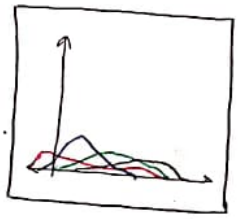


Date wise average

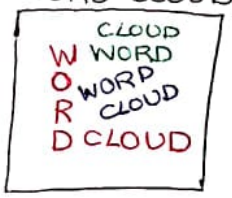


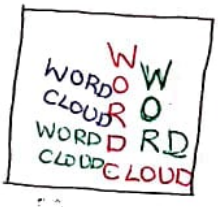
WORLD MAP



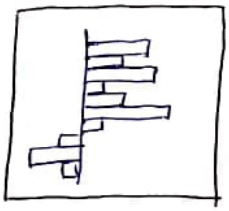


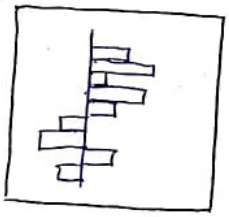
WORD CLOUD





STATE WISE ANALYSIS





INFORMATION

DATE: 29th MAY, 2019

SHEET: #4

TOPIC: SENTIMENT ANALYSIS.

COMPONENTS

1) Single page direct comparison representation

2) World map with label used instedd of chloroplet-th to locate tweet coordinates.

3) Pie chart used to show polority.

DISCUSSION.

POSITIVE:

1) Complete comparison.

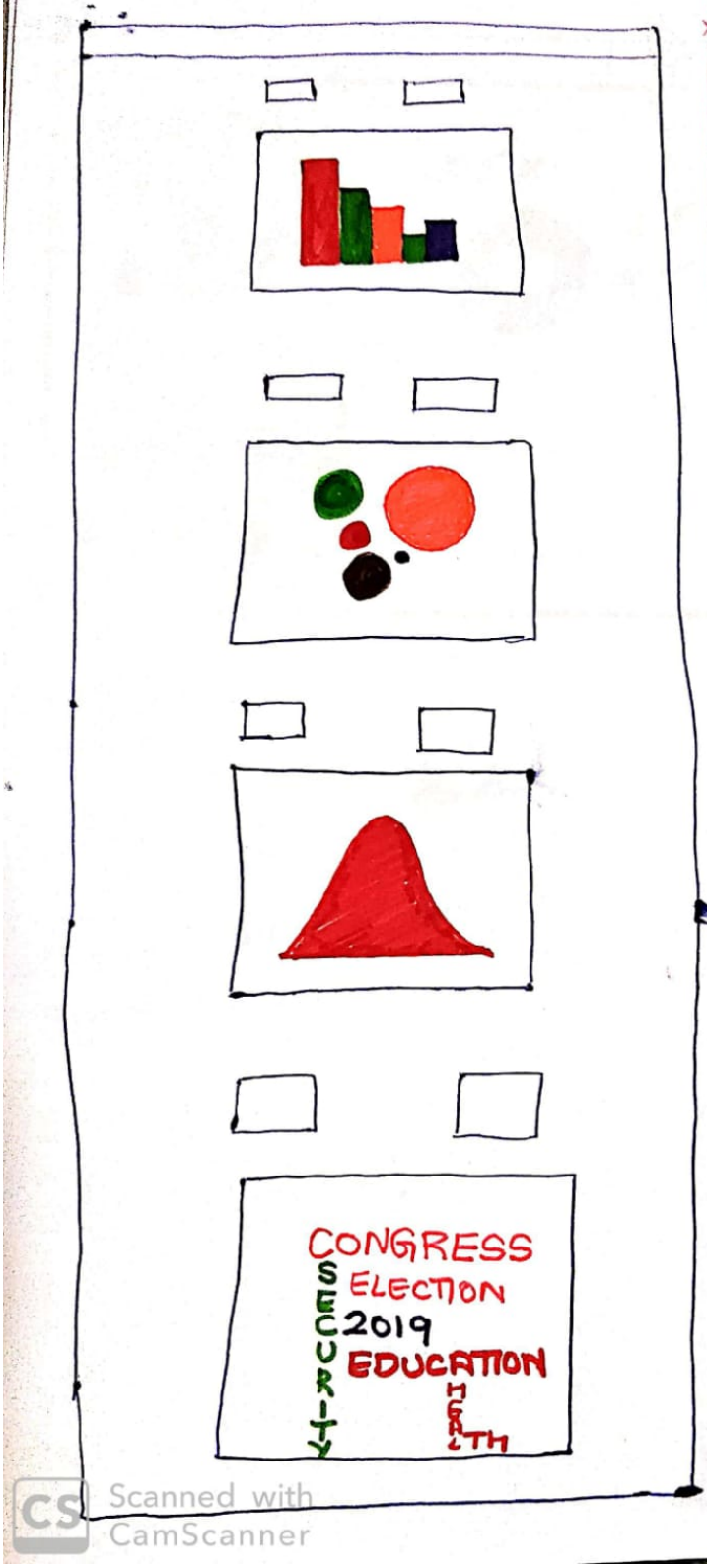
2) World map with label shows more clear data.

NEGATIVE:

1) In pie-chart actual values are not clear for similar amount.

2) Too much information on a single page.

FINAL DESIGN



INFORMATION

TOPIC : SENTIMENT ANALYSIS

PAGE : 5

REALIZATION

COMPONENT

→ Single page scroll down interface.

→ Option to switch the party on top of each plot.

→ A small description about each plot will be added after each plot.

DETAIL :

Following plots will be used ~~to~~ for comparison.

1) Text comparison
WORD CLOUD.

2) Location comparison
INDIA MAP

3) SENTIMENT COUNT
HISTOGRAM

4) ~~Den~~ Distribution of sentiment
DENSITY PLOT

5) Online Impression
BUBBLE CHART