Analyzing Political Sentiment using Twitter Data

Technical Report · May 2018

Sandip Roy
Brainware University

1. Introduction

As because of the massive growth of user-created data in the recent WWW websites, people from various backgrounds tweet massive amount of textual remarks deliberating their thoughts in a different perspective of their emotions and public to everyone. Natural Language Processing (NLP) can be categorized into opinion and text mining. This technique is helped for isolating the opinions of posting on various social media platforms like Twitter, Reddit, and Facebook etc. In today's world text or opinion, mining is helpful for judging public views regarding a newly released item, movie, song, book, etc. [1]. It also differentiated among positive, negative and neutral opinion and recommendations. It also becomes a common practice for the common people to pose their expressions towards the political leader on social media. Different reporters have been taking an interview with the political leader to know their views and communicate with the people through TV program, YouTube, etc. People express their opinions with each other regarding the political talks which ran on the TV show. It is very expensive and time-consuming task to search people's opinions via surveys and polls.

Now various social media sites (like Twitter, Reddit, Facebook, etc.) are extensively used by the public when they can share their opinions publicly [2]. One of important microblogging site receiving around 500 million tweets every day where the daily limit of each user is 2,400 tweets and 140 characters every tweet [3]. Hence, Twitter is one of the relevant platforms where each people can connect with different communities and express their opinions loud and clear.

Sentiment analysis is a type of data mining process that determines the public opinion through NLP. It is the process of classifying opinions into three categories like "positive", or "negative" or "neutral" [4]. This data quantifies the public's reactions toward certain people, communities, and political discourses which divulge the contextual polarity of the information.

Many research already organized on political sentiment analysis on Twitter data. The main aim of the research is to predict the election result based on people's opinions. Popular debate TV show between Barack Obama and John McCain called "Hack the Debate". In this TV show, the audience freely posted comments on Twitter. It has become an admissible communication platform in the political arena, on basis of the successful victory of Barack Obama. Hence, political sentiment analysis is a common practice for analyzing the people's opinion towards the political campaign.

Due to the second-most populous country and the most populous democracy in the world, India political situation is most fluctuating. Every step of the ruling party would have several views of the oppositions. But today common people post their opinions on the social media regarding every political step (like demonetization of all Indian five hundred and one thousand currency of the Mahatma Gandhi Series).

Therefore our aim is to analyze the emotion of web users concerning every political party, their leaders and their steps based on tweets on the social media [5]. The detailed sections of the thesis are as follows. Section 2 is described literature Survey. Section 3 deals with on data preparation along with our corpus analysis. Experimental setup and result analysis part are discussed in section 4 and the section 5 draws conclusions and examines the possibilities of next future.

2. Literature Review

Literature survey can be categorized into two different subsections, Traditional sentiment analysis and examine related to the people's political judgment.

2.1 Traditional Sentiment Analysis

There are various research papers published on sentiment analysis.

Tumey's unsupervised learning algorithm examined Pointwise Mutual Information (PMI) for estimating the opinion inclination of phrases [6]. In opposed to it, Pang, Lee, and Vaithyanathan suggested that three machine learning methods such that, Support Vector Machines, Naive Bayes, and Maximum Entropy Classification don't function as well on the classification of sentiment or on traditional topic-based segregation [7].

Chesley et al. presented that how a particular blog expressed subjectivity vs. objectivity and also how a post expressed positive or negative polarity [8]. Godbole et al. researched on recent events reported in newspapers and blogs which expressed opinions of news entities (people, places and things). Their assigning scores representing sentiment to each separate entity in the lexicon [9]. Early research contributed to a longer document such as movie review and blogs.

Bermingham et al. presented in his research on microblogs where they examined the hypothesis that easily classifies the opinion in small-scale documents than in larger scale documents [10]. They experimented with the microblogs from Twitter, blog posts, micro-reviews from blippr and movie reviews and culminated that it is easier to identify sentiment from microblogs. There has been some research on sentiment analysis on Twitter. Pak and

Paoubek suggested that microblogging has become a popular tool for expressing opinions of people. Their sentiment classifier can determine the sentiments of a given text [11]. Davidov also examined on data from Twitter [12]. Their supervised sentiment classification framework performed on fifteen smileys as sentiment labels and fifty Twitter tags which identified and classified of various sentiment types of short texts.

Bakliwal et al. focused on the task of tweet sentiment identification using a corpus of pre-annotated tweets [13]. They have used unigram and bigram features and presented a sentiment scoring function which boosts the classification accuracy [13].

2.2 Political Sentiment Analysis

On-demand interest in the online political sentiment analysis is to predict the result of the election.

Tumasjan et al. focused in his research on the German federal election in 2009 [14]. Their investigation identified that the Twitter became as a platform for analyzing sentiment and also predicting the outcome of the election [14]. They examined on around one lakh political tweets identifying either a politician or a political party. They used LIWC2007 tool for extracting sentiment from the tweets [14, 15]. LIWC is accurate text analysis software developed to reveal people thoughts, emotions, cognitive and personality using text samples [15]. They concluded that the number of tweets is directly proportional to the chances of winning the election.

O'Connor investigated on the people's remark measured from polls with opinion measured from microblogging sites [16]. They analyzed various political sentiment from 2008 - 2009, and result in the correlation between the frequencies of sentiment word in simultaneous with the twitter tweets.

Choy et al. proposed an application of online sentiment analysis to predict the vote percentage for each of the candidates in the Singapore presidential election of 2011 [17]. Wang et al. presented a real-time sentiment application system for U.S. presidential election of 2012 based on political tweets extracted from Twitter [18]. Ringsquandl and Petković worked on the campaign topics of presidential candidates of the Republican Party in the USA. Their research, they introduced the amalgamation of the frequencies of noun phrases and their PMI measure with a constraint on aspect extraction and concluded that the semantic relationship between politicians and their topics holds. Their result has also improved the accuracy of the aspect extraction [19].

Elghazaly, Mahmoud, and Hefny also focused on Egypt Presidential Elections '12 based on Arabic text classification based on WEKA application. They concluded that the Naïve Bayesian method gives the highest accuracy with the lowest error rate [20]. In general state election in 2016, Sharma and Moh researched on Hindi Twitter for predicting of Indian state election. They have used 42,235 tweets using Twitter Archiver tool to extract tweets in Hindi language and examined by Support Vector Machine (SVM), Dictionary Based and Naive Bayes algorithm which classified into three categories like "positive", or "negative" or "neutral" [21]. The objective of our research is to analyze peoples' sentiment for determining the results of assembly election of Gujarat in 2017 using deep learning method tools available as web service by ParallelDots Inc [22].

3. Data Preparation

The data has been collected during November 09, 2017 to January 07, 2018 using Twitter's streaming API [23] (See Appedix). We have collected around 1,000 tweets mentioning two verified Twitter accounts named @vijayrupanibjp and @BharatSolankee respectively, the leader of Gujarat Legislative Assembly election, 2017 as mentioned on Wikipedia [24].

3.1 Twitter Data Analysis

We explore the time series analysis of tweets over given time period and we visualize that most of the tweets are from Twitter for Android followed by Twitter for iPhone and Twitter Web Client source which is depicted on figure 1.

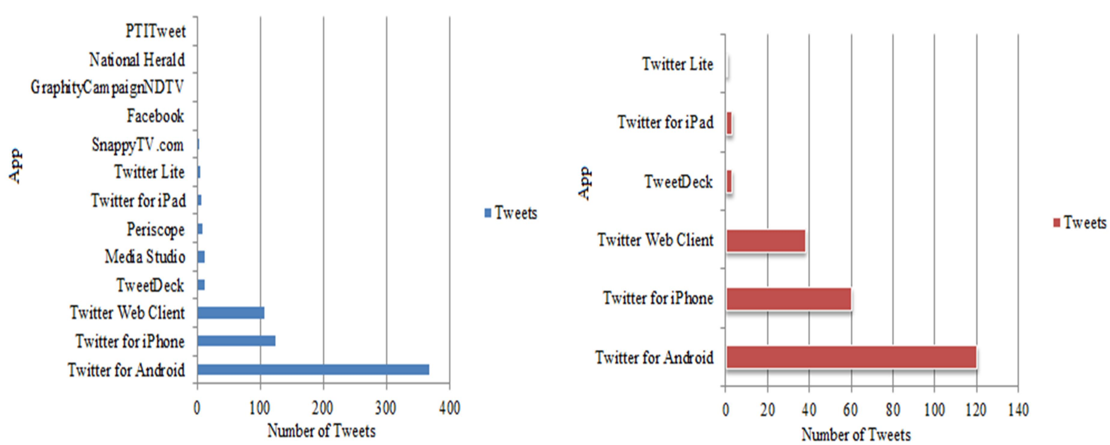

Figure 1. Number of Tweets extracted from different sources

We also explore that on which time of a day maximum retweets is contributed from which Twitter account in our corpus is shown below in Fig. 2 and Fig. 3 respectively.

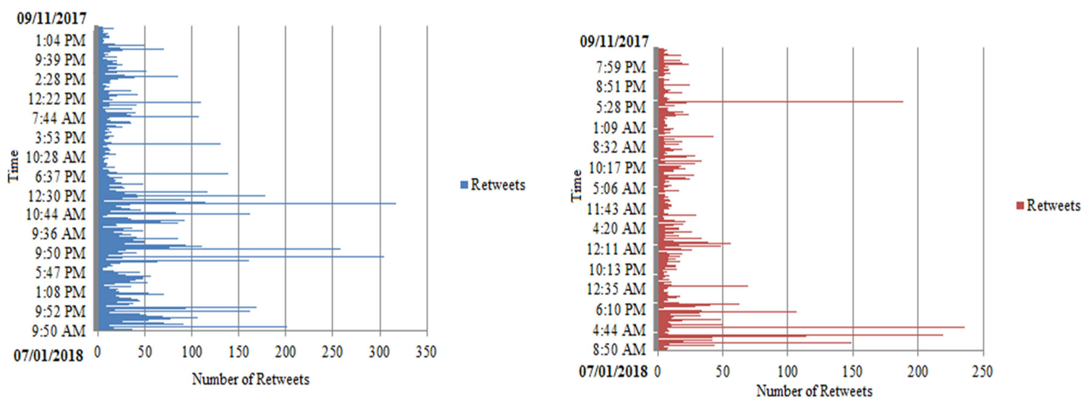

Figure 2. Number of Retweets vs. Time

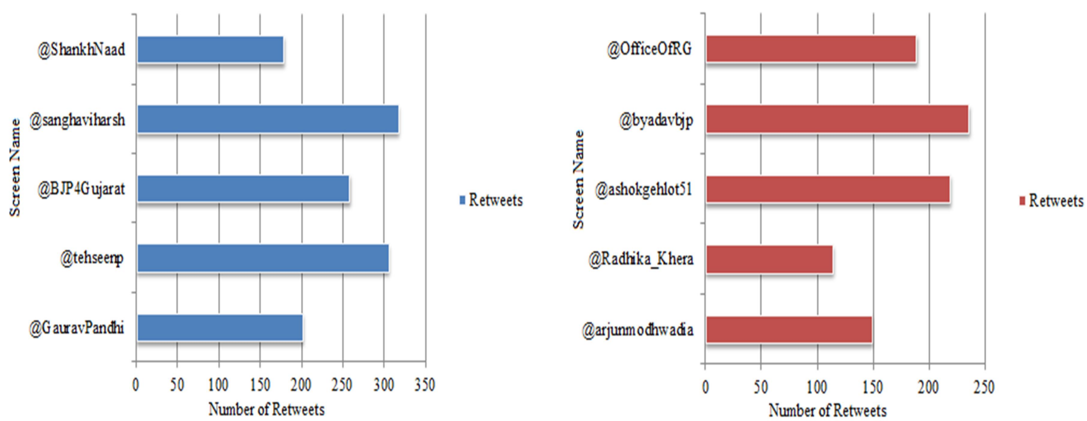

Figure 3. Number of Retweets vs. Screen Name

3.2 Data Processing

Before applying any of the sentiment or emotion analyzing method [25, 26], we perform data preprocessing. Data preprocessing is a major step in text mining that produces a higher quality of text classification and reducing computational complexity [27]. The following points have mentioned as data preprocessing steps of the tweet.

– Remove all hashtags (e.g. #topic), screen name (e.g. @username) and all URLs (e.g. www.xyz.com)

– Remove all punctuations, symbols, and numbers

– Replace any non UTF-8 by space

– Remove Stopwords

– Substitute all the emoticons with their sentiment

– Change text to lowercase

– Replace words with their stems or roots

4. Experimental Setup and Result Analysis

We performed the Emotion Analysis using syuzhet in CRAN package which is based on NRC Emotion Lexicon [28]. Figure 4 and figure 5 show the distribution of emotions of Vijay Rupani and Bharatsinh Madhavsinh Solanki during the Gujarat Legislative Assembly election, 2017.

In figure 4, more than 300 tweets express positive sentiment and around 160 tweets indicate negative sentiment whereas in figure 5, more than 100 tweets express positive and around 70 tweets represent negative sentiment. In figure 4, more than 200 tweets express trust whereas in figure 5 around 90 tweets express trust as an emotion. As can be seen, the disgust emotion in figure 4 is second least emotion score compare to the figure 5. Similarly, joy emotion is placed better in figure 4 than the emotion score in figure 5.

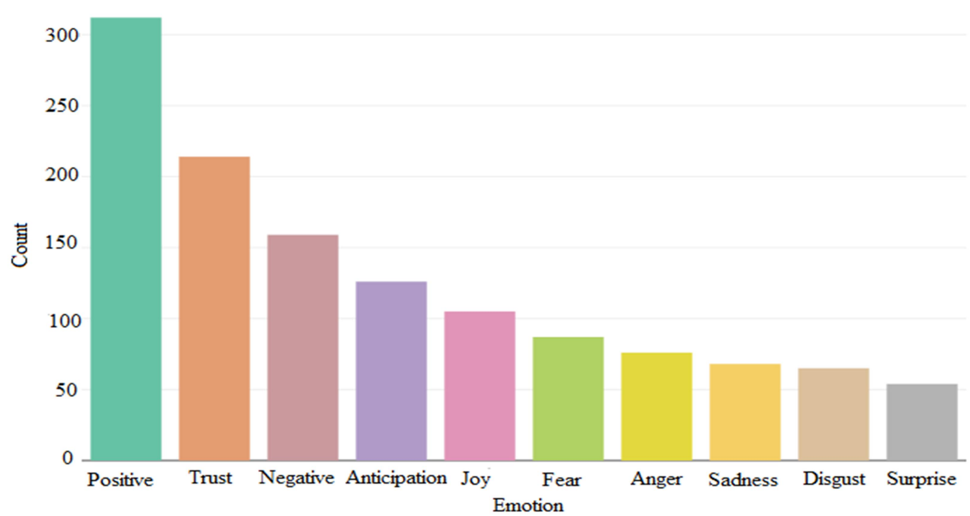

Figure 4. Emotion distribution of Vijay Rupani for Gujarat Legislative Assembly election, 2017

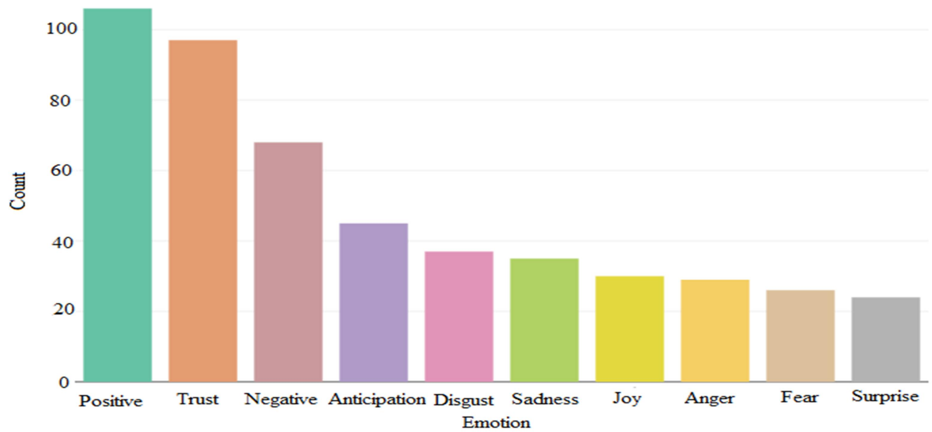

Figure 5. Emotion distribution of Bharatsinh Madhavsinh Solanki for Gujarat Legislative Assembly election, 2017

In figure 6 shows sentiment score using ParallelDots AI API which gives positive, negative and neutral sentiment for the event with an accuracy of 88% [29]. This figure 6, Vijay Rupani, the elected candidate from BJP with 55% positive tweets has a big possibility of triumph the election.

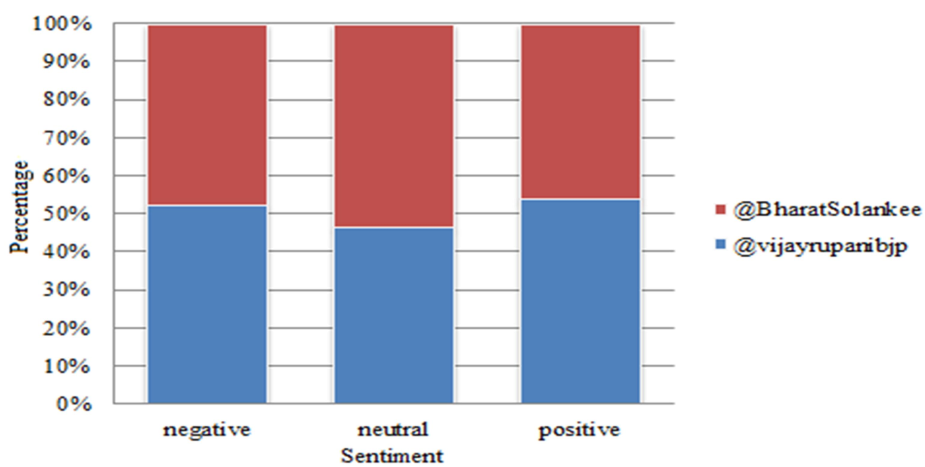

Figure 6. Percentage of number negative, neutral and positive tweets using ParallelDots AI APIs

5. Conclusion and Future Scope

Only 12.18% of people of the total population in India have communicated using English language and most of the common people express their opinions using native language [30]. Therefore analyzing sentiment using the only English language isn't worthy of every time. Our research will be more robust when we will implement the lexicon of different native languages used in India (e.g. Hindi, Gujarati etc.) [21]. Furthermore, we will also extend our research using Latent Dirichlet Allocation (LDA) based topic model and also identifies the correlated topic across a different category for the event [26].

6. Declaration

The authors declare and solemnly affirm that this research has neither been funded by any political or religious groups nor are the authors in any way affiliated to any institutions with direct or indirect access to groups with biased interests. This research work has been carried out exclusively and independently by the authors in the interests of technology and progress of science in sentiment analysis and related fields.

References

1. Mohey El-Din Mohamed Hussein, D.: A survey on sentiment analysis challenges. J. of King Saud Univ. – Engg. Sci., 1 – 9 (2016)
2. Haddia, E., Liua, X., Shib, Y.: The Role of Text Pre-processing in Sentiment Analysis. Info. Tech. and Quant. Manag. (ITQM2013), Proc. Comp. Sci. 17, 26 – 32 (2013)
3. Mamgain, N., Mehta, E., Mittal, A., Bhatt, G.: Sentiment Analysis of Top Colleges in India using Twitter Data. Int. Conf. on Comp. Tech. in Info. and Comm. Tech. 1 – 6 (2016)
4. Medhat, W., Hassan, A. Korashy, H.: Sentiment analysis algorithms and applications: A survey. A. S. Engg. J. 5, 1093–1113 (2014)
5. Shamma, D. A., Kennedy, L., Churchill, E. F.: System and method for using topic messages to understand media relating to an event. US8396874 B2, 1 – 17 (2013)
6. Turney, P. D.: Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In Proc. of the 40th Annual Meeting on Asso. for Comp. Ling. Philadelphia, 417 – 424 (2002)
7. Pang, B., Lee, L. Vaithyanathan, S.: Thumbs up? Sentiment Classification using Machine Learning Techniques. In Proc. of the conf. on Emp. Meth. in Natural Lang. Proc. 10, 79 – 86 (2002)
8. Chesley, P.: Using Verbs and Adjectives to Automatically Classify Blog Sentiment. In Proc. of AAAI-CAAW-06. The Spring Symp. on Comp. Appro., 1 – 3 (2006)
9. Godbole, N., Srinivasaiah, M. Skiena, S.: Large-Scale Sentiment Analysis for News and Blogs. In Proc. of the Int. Conf. on Web. and Soc. Med. (ICWSM), Colorado, USA, 1 – 6 (2006)
10. Bermingham, A., Smeaton, A. F.: Classifying sentiment in microblogs: is brevity an advantage? In Proc. of the 19th ACM int. conf. on Inform. and Know. Manag. 1833 – 1836 (2010)
11. Pak, A., Paroubek, P.: Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In Proc. of the Seventh Int. Conf. on Lang. Res. and Eval. (LREC'10). 1320 – 1326 (2010)
12. Davidov, D., Tsur, O., Rappoport, A: Enhanced Sentiment Learning Using Twitter Hashtags and Smileys. In Proc. of the 23rd Int. Conf. on Comp. Ling.: Posters, 241-249 (2010)

13. Bakliwal, A., Arora, P., Madhappan, S., Kapre, N., Singh, M. Varma, V.: Mining Sentiments from Tweets. In Proc. of the WASSA’12 in conj. with ACL’12. 11 – 18 (2012)
14. Tumasjan, A., Sprenger, T. O., Sandner, P. G., Welpe, I. G.: Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment. In Proc. of the Int. Conf. on Web. and Soc. Med. 178 – 185 (2010)
15. Pennebaker, J. W., Chung, C. K., Ireland, M., Gonzales, A., Booth, R. J.: The Development and Psychometric Properties of LIWC2007. Tech. Report, Austin, Texas, 1 – 22 (2007)
16. O’Connor, B., Balasubramanyan, R., Routledge, B. R., Smith, N. A.: From tweets to polls: Linking text sentiment to public opinion time series. In Proc. of the Int. Conf. on Web. and Soc. Med. (ICWSM), 122 – 129 (2010)
17. Choy, M., Cheong, M. L. F., Laik, M. N., Shung, K. P.: A Sentiment Analysis of Singapore Presidential Election 2011 using Twitter data with Census Correction. Research Collection Sch. of Info. Sys. 1 – 12 (2012)
18. Wang, H., Can, D., Kazemzadeh, A., Bar, F., Narayanan, S.: A System for Realtime Twitter Sentiment Analysis of 2012 U.S. Presidential Election Cycle. ACL '12 Proc. of the ACL 2012 Sys. Demons. 115 – 120 (2012)
19. Ringsquandl, M., Petković, D.: Analyzing Political Sentiment on Twitter. Association for the Adv. of Artif. Intell. 40 – 47 (2013)
20. Elghazaly, T., Mahmoud, A., Hefny, H. A.,: Political Sentiment Analysis Using Twitter Data. In Proc. of the Int. Conf. on Inter. of things and Cloud Comp., 11 (2016)
21. Sharma, P., Moh, T-S.,: Prediction of Indian Election Using Sentiment Analysis on Hindi Twitter. 2016 IEEE Int. Conf. on Big Data (Big Data). 1966 – 1971 (2016)
22. Sharma, Y., Mangat, V., Kaur, M.: A Practical Approach to Sentiment Analysis of Hindi Tweets. In Proc. of the 2015 1st Int. Conf. on Next Gen. Comp. Techno. (NGCT). 677 – 680 (2015)
23. Agarwal, A.: How to Save Tweets for any Twitter Hashtag in a Google Sheet. Digi. Insp. (2015)
24. Gujarat Legislative Assembly election, 2017. Wikipedia. (2017)
25. Kharde, V. A., Sonawane, S.S.: Sentiment Analysis of Twitter Data: A Survey of Techniques. Int. J. of Comp. App. 139(11), 5 – 15 (2016)
26. Niyogi, M., Pal, A. K. Discovering conversational topics and emotions associated with Demonetization tweets in India. ArXiv: 1711.04115v1 [cs.CL]. 1 – 6 (2017)

27. Vijayarani, Dr. S., Ilamathi, J., Nithya, M.: Preprocessing Techniques for Text Mining - An Overview. Int. J. of Com. Sci. & Comm. Net. 5(1), 7-16 (2015)
28. Mohammad, S. M., Kiritchenko, S. Using Hashtags to Capture Fine Emotion Categories from Tweets. Sp. issue on Semantic Analy. in Soc. Med., Comp. Intell. 1 –22 (2013)
29. Sharma, G.: Extract the context: Sentiment Analysis and Opinion Mining. ParallelDots. (2017)
30. List of languages by number of native speakers in India. Wikipedia. (2017)

Appendices

Appendix 1: How to extract tweets using R language

Prerequisites

1. You have already installed R and are using RStudio or install R using https://cran.r-project.org/bin/windows/base/.

2. In order to extract tweets, you will need a Twitter application and hence a Twitter account. If you don't have a Twitter account, sign up using https://twitter.com/signup.

3. Use your Twitter login ID and password to sign in at Twitter Developers using https://apps.twitter.com/.

Steps for creating Twitter Apps

1. Create a new application

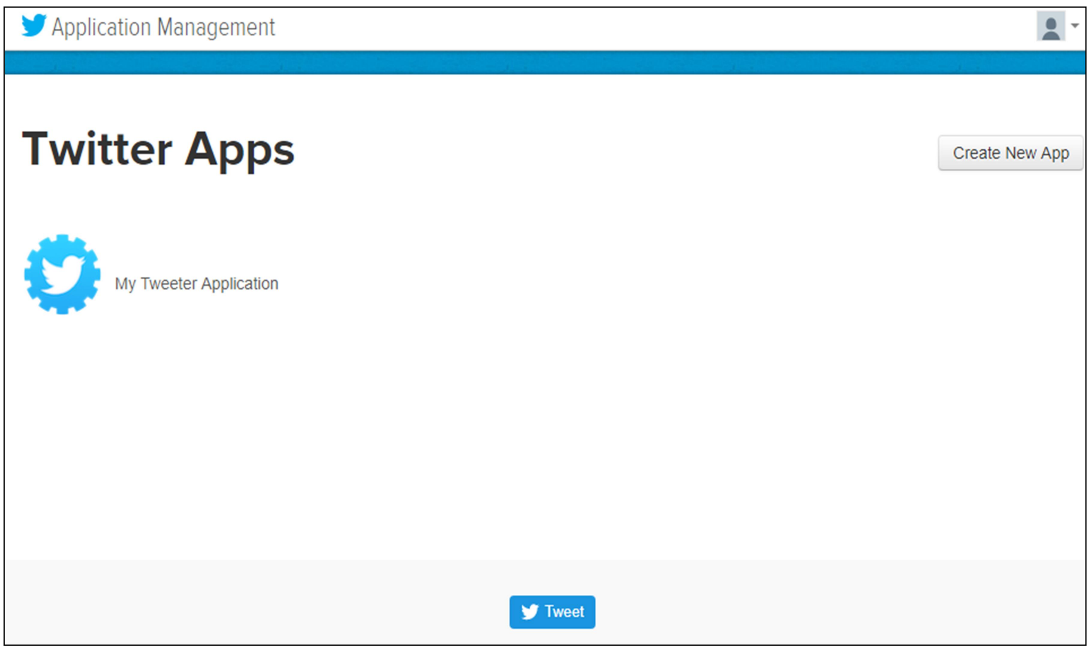

Figure A.1.1 Application Management Menu of Twitter

2. Fill out the new app form. Names should be unique, i.e., no one else should have used this name for their Twitter app. Give a brief description of the app. You can change this later on if needed. Enter your website or blog address. Callback URL can be filled by http://127.0.0.1:1410. Then click the "Create your Twitter application" button.

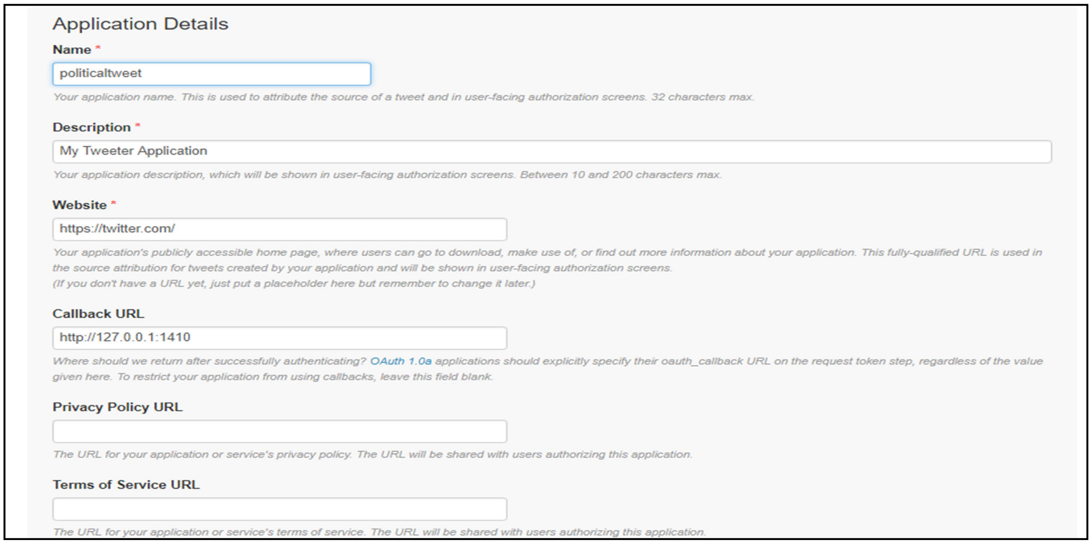

Figure A.1.2 Create an application Menu of Twitter

3. Scroll down and click on "Create my access token" button. Note the values of consumer key and consumer secret and keep them secure for future use. If anyone was to get these keys, they could effectively access your Twitter account.

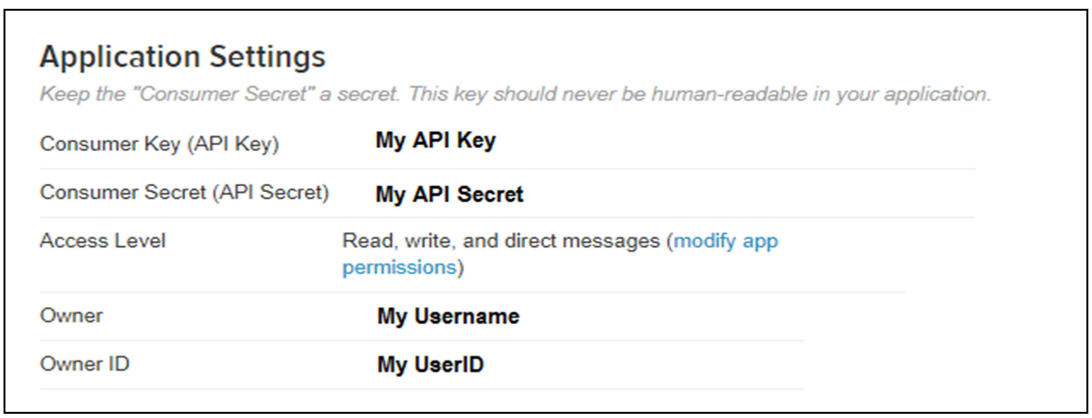

Figure A.1.3 Application Settings Menu of Twitter

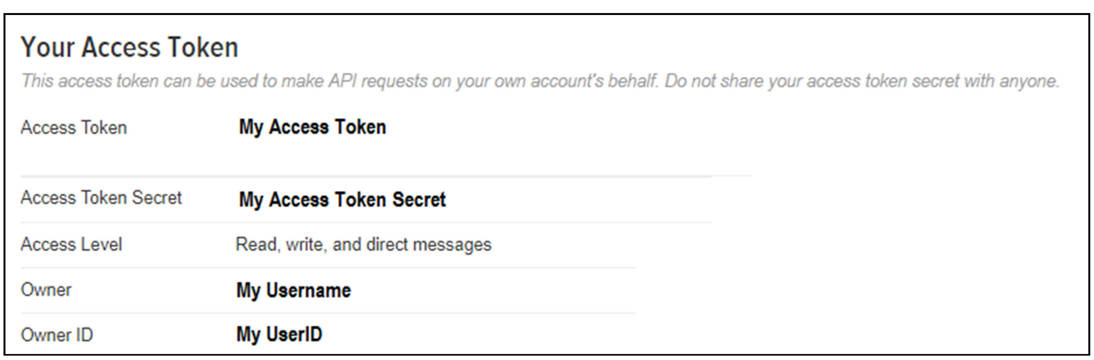

Figure A.1.4 Access Token Menu of Twitter

4. Run the below script on RGUI.

```
install.packages("twitteR")
install.packages("RCurl")
require(twitteR)
require(RCurl)
# User API Key
api_key ← "XX"
# User API Secret
api_secret ← "XX"
"# User Access Token
access_token ← "XX
# User Access Token Secret
access_token_secret ← "XX"
setup_twitter_oauth(api_key, api_secret, access_token, access_token_secret)
#Search BJP tweets for an example
mytweet ← searchTwitter("#hashtag", n = 10, lang = "en")
#Display tweets
mytweet
```

Appendix 2: How to extract tweets using Twitter Archiver Tool

Note: It was collected using Google Spreadsheet which built the connection to Twitter using Google script by finding the key details from a Twitter account and authorize the applications. After that it can automatically import all the users' recent tweets into the Spreadsheet. The below steps are required to connect for extracting tweets using Twitter Archiver Tool.

Step 1: Go to Google drive and click on the "New" tab to open the Blank spreadsheet.

Step 2: Go to the Add-ons menu, choose Twitter Archiver Tool and select the Authorize menu

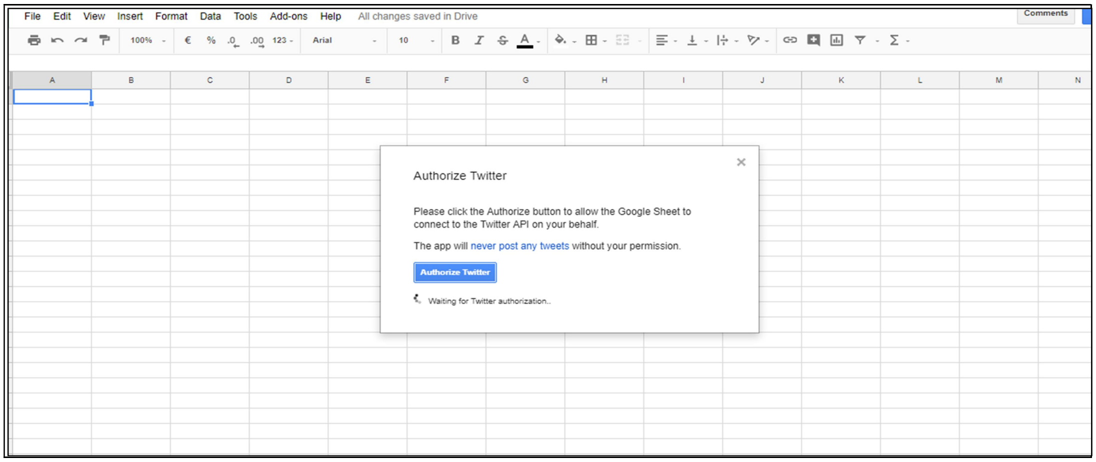

Figure A.2.1 Authorize Twitter menu to allow Google Sheet to connect to Twitter API

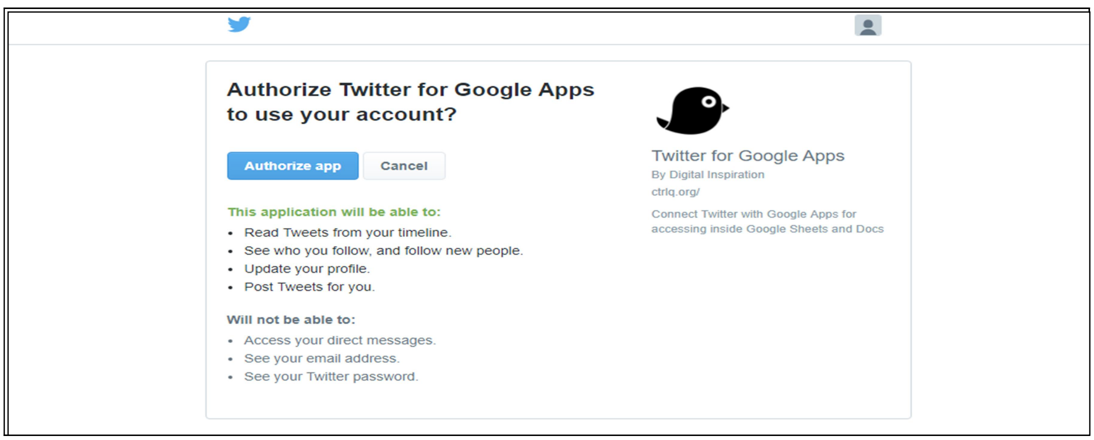

Figure A.2.2 Authorize Twitter for Google Apps to use your Twitter account

Step 3: Allow the Google Sheets to authorize Twitter on your behalf.

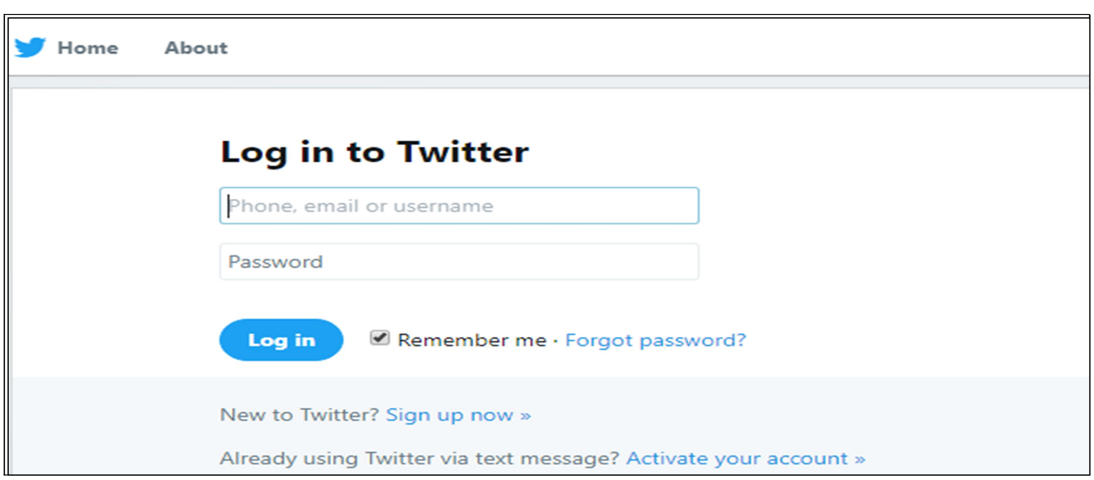

Figure A.2.3 Log-in to twitter account using username and password

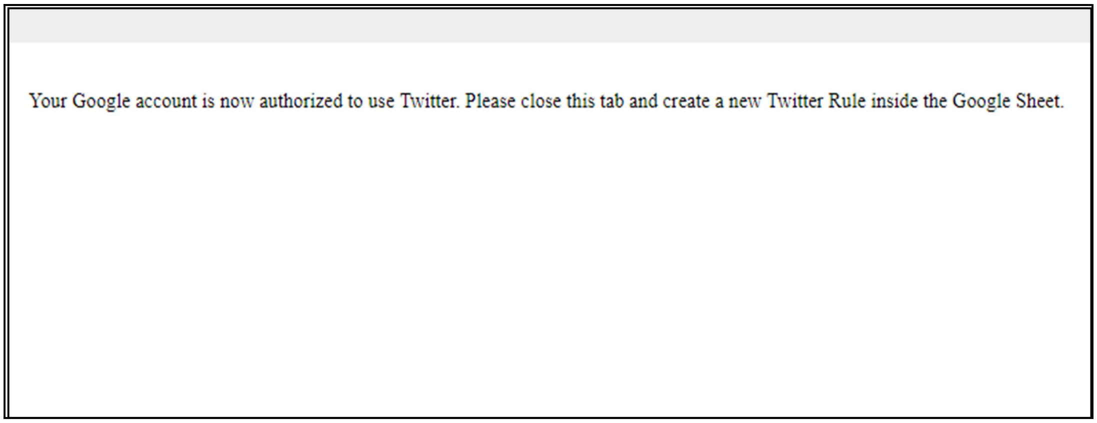

Figure A.2.4 Screen display after successful authentication

Step 4: After successful authentication, go to the Twitter Archiver menu again and create a new Search Rule.

Note: Steps for extract tweets using R language (See Appendix 1)

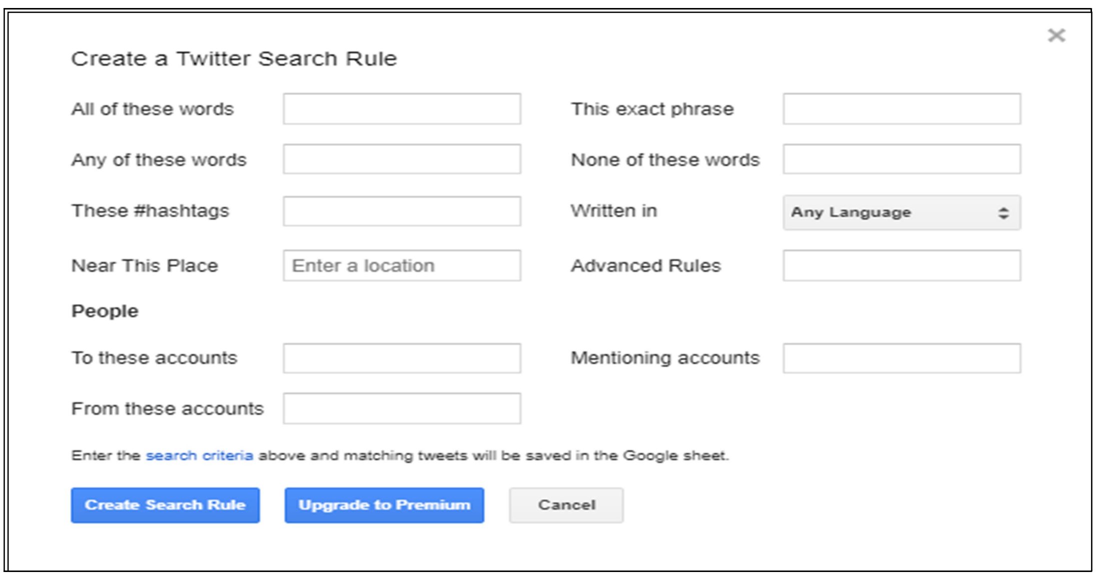

Figure A.2.5 Twitter Search Rule window

2	Date	Screen Name	Full Name	Tweet Text	Tweet ID	App	Followers	Follows
3	11/9/2017 9:50:29	@sanjayezhava	Sanjay Ezhava IN	@PiyushGoyal @RailMinIndia @drmmumbaicr Seems Liq	928477368783577088	Twitter for iPhone	11293	1837
4	11/9/2017 10:33:11	@ramanmann1974	RamanS Mann	#SEBI charges Gujarat Chief Minister @vijayrupanibjp's H	928488115957743616	Twitter Web Client	2768	72
5	11/9/2017 11:07:41	@RuchiraC	Ruchira Chaturvedi	SEBI penalises Gujarat CM @vijayrupanibjp owned comp	928496796271239168	Twitter for Android	3974	433
6	11/9/2017 12:06:03	@yogen198	Yogen Bhawsar	@BJP4India @AmitShah @DrGPradhan @INCIndia @INC(	928511485168443392	Twitter for Android	3025	1883
7	11/9/2017 12:42:51	@dangarbharat	Bharat Dangar	#Vadodara welcoming Honourable Chief Minister @vijay	928520746661326850	Twitter for Android	33115	494
8	11/9/2017 13:18:53	@das_prashant777	Dr Prashant Das	@DhartiMajithiya @PoonambenMaadam @narendramod	928529814230675456	Twitter for Android	2688	1095
9	11/9/2017 14:26:49	@srirambjp	Sriram	Muslims turn up in large numbers at Rupani rally. @vijay	928546911530840064	Twitter for iPhone	24645	1516
10	11/9/2017 15:30:35	@Barmer_Harish	Harish Chaudhary	Another case of corruption, another malicious BJP man, a	928562955641274371	TweetDeck	4544	78
11	11/9/2017 16:02:57	@GauravPandhi	Gaurav Pandhi	What a fraud !! Gujarat CM @vijayrupanibjp has been per	928571101529104384	Twitter for Android	49700	658
12	11/9/2017 16:20:24	@rachitseth	Rachit Seth	Congress party demands the resignation of Gujarat CM @	928575491908321281	Twitter for iPhone	41059	2690
13	11/9/2017 16:45:41	@dangarbharat	Bharat Dangar	Glimpses of today's #Gujarat_Gaurav_Mahasampark_Abh	928581856630415360	Twitter for Android	33115	494
14	11/9/2017 16:49:42	@Trollmodii	Narendra Godi	@vijayrupanibjp Vijuuu now u r finally my boy.... Good jo	928582866014842880	Twitter for Android	571	43
15	11/9/2017 17:47:43	@VPra52	Vinay Prasad	SEBI has imposed total penalties worth Rs 6.9 crores on \	928597465829335040	Twitter for Android	5124	3062
16	11/9/2017 18:14:22	@pvssarma	pvs sarma	There's no leader in @BJP4Gujarat against who allegation	928604176027807745	Twitter for Android	4565	239
17	11/9/2017 18:32:04	@sureshnakhua	Suresh Nakhua	So @bsindia does a hit job on @vijayrupanibjp . Does a st	928608627341934597	Twitter for Android	70651	2770

Figure A.2.6 Import tweets to the Google Spreadsheet

Appendix 3: How to preprocess tweets using R language

1. Download the search tweets from Google drive which are extracted by Twitter Archiver Tool.

2. Run the below script on RGUI

```
s <- read.csv("file name")
x <- list(s[,4])
# Remove at people mentioned preceding by ‘#’ or ‘@’
x = gsub("#\\w+", "", s[,4])
x = gsub("@\\w+", "", x)
# Replace any non UTF-8 by space
x=iconv(x, "UTF-8", "UTF-8",sub='')
# Remove punctuation
x = gsub("[[:punct:]]", "", x)
#Remove alphanumeric character
x = gsub("[^[:alnum:]]", " ",x)
# Remove numbers
x = gsub("[[:digit:]]", "", x)
# Remove html links
x = gsub("http\\w+", "", x)
# Remove unnecessary spaces
x = gsub("[ \t]{2,}", "", x)
x = gsub("^\\s+|\\s+$", "", x)
```

```
#Convert text to tolower
x=tolower(x)
# Write x to csv file
write.csv(x, "output file name")
```

Appendix 4: How to analyze emotion using syuzhet in CRAN package

1. Download the search tweets from Google drive which are extracted by Twitter Archiver Tool.

2. Run the below script on RGUI

```
library(syuzhet)
library(plotly)
library(tm)
library(wordcloud)
# Read the file name
s <- read.csv("file name")
Timestamp <- list(s[,1])
Account <- list(s[,2])
Tweets <- list(s[,4])
Retweet <- list(s[,9])
Tweets = gsub("#\\w+", "", s[,4])
Tweets = gsub("@\\w+", "", Tweets)
# Replace any non UTF-8 by space
Tweets = iconv(Tweets, "UTF-8", "UTF-8",sub=")
# Remove punctuation
Tweets = gsub("[[:punct:]]", "", Tweets)
#Remove alphanumeric character
Tweets = gsub("[^[:alnum:]]", " ", Tweets)
# Remove numbers
Tweets = gsub("[[:digit:]]", "", Tweets)
# Remove html links
Tweets = gsub("http\\w+", "", Tweets)
# Remove unnecessary spaces
Tweets = gsub("[ \t]{2,}", "", Tweets)
Tweets = gsub("^\\s+|\\s+$", "", Tweets)
```

```
#Convert text to tolower
Tweets = tolower(Tweets)
Outfile <- list(Timestamp, Account, Tweets, Retweet)
write.csv(Outfile, "Output preprocess file name")
l <- as.Date(as.character(s[,1]),  format = "%m/%d/%Y")
m <- l[!duplicated(l)]
syuzhet <- get_sentiment(Tweets, method="syuzhet")
bing <- get_sentiment(Tweets, method="bing")
afinn <- get_sentiment(Tweets, method="afinn")
nrc <- get_sentiment(Tweets, method="nrc")
sentiments <- data.frame(syuzhet, bing, afinn, nrc, l)
write.csv(sentiments, "Output sentiment file name")
#get the emotions using the NRC dictionary
emotions <- get_nrc_sentiment(Tweets)
emo_bar = colSums(emotions)
emo_sum = data.frame(count=emo_bar, emotion=names(emo_bar))
emo_sum$emotion = factor(emo_sum$emotion,
      levels=emo_sum$emotion[order(emo_sum$count, decreasing = TRUE)])
# plot the different sentiments from different methods
plot_ly(sentiments, x=~l, y=~syuzhet, type="scatter", mode="jitter", name="syuzhet") %>%
  add_trace(y=~bing, mode="lines", name="bing") %>%
  add_trace(y=~afinn, mode="lines", name="afinn") %>%
  add_trace(y=~nrc, mode="lines", name="nrc") %>%
  layout(title="Recent sentiments of Vijay Rupani in India",
         yaxis=list(title="score"), xaxis=list(title="date"))
# Visualize the emotions from NRC sentiments
plot_ly(emo_sum, x=~emotion, y=~count, type="bar", color=~emotion) %>%
  layout(xaxis=list(title=""), showlegend=FALSE,
         title="Distribution of emotion categories for Vijay Rupani (21 Nov - 1st Dec 2017)")
# Comparison word cloud
all = c(
  paste(Tweets[emotions$anger > 0], collapse=" "),
  paste(Tweets[emotions$anticipation > 0], collapse=" "),
  paste(Tweets[emotions$disgust > 0], collapse=" "),
```

```
  paste(Tweets[emotions$fear > 0], collapse=" "),
  paste(Tweets[emotions$joy > 0], collapse=" "),
  paste(Tweets[emotions$sadness > 0], collapse=" "),
  paste(Tweets[emotions$surprise > 0], collapse=" "),
  paste(Tweets[emotions$trust > 0], collapse=" ")
)
all <- removeWords(all, stopwords("english"))
# create corpus
corpus = Corpus(VectorSource(all))
# create term-document matrix
tdm = TermDocumentMatrix(corpus)
# convert as matrix
tdm = as.matrix(tdm)
tdm1 <- tdm[nchar(rownames(tdm)) < 11,]
# add column names
colnames(tdm) = c('anger', 'anticipation', 'disgust', 'fear', 'joy', 'sadness', 'surprise', 'trust')
colnames(tdm1) <- colnames(tdm)
comparison.cloud(tdm1, random.order=FALSE,
                 colors = c("#00B2FF", "red", "#FF0099", "#6600CC", "green", "orange", "blue",
       "brown"),
                 title.size=1, max.words=250, scale=c(2.5, 0.4),rot.per=0.4)
```