DATA INTENSIVE COMPUTING

Sentiment Analysis on US Presidential Elections 2016

Ramanpreet Singh Khinda | DIC 587 | March 5, 2016
rkhinda@buffalo.edu
50169622

Problem 5: Twitter Stream processing and R Shiny

This is an Election Season and everyone is talking about how will going to be the next President of US and many data analytics companies are gauzing the Social Network data to predict the future.

Since most users take on twitter to express their sentiments on these topics. We collected data on top candidates for Democratic and Republican party. Based on our Analysis we tried to predict the future ☺ and claimed our next president.

Let's walk through the process step by step

Step 1: Data Collection

First we collected data on US Elections for over 1 week (From 28 Feb to 5 March)

Since this is a live ongoing topic we have also opened a live streaming connection to the twitter api and analyzing each tweet.

```
34      filterStream(file.name="Tweets_Hillary_Clinton.json",
35                   track=c("#ImWithHer","#hillaryclinton", "#WhyImNotVotingForHillary"),
36                   tweets=1000, oauth=credential, timeout=10, lang='en' )
37      filterStream(file.name="Tweets_Bernie_Sander.json",
38                   track=c("#bernie2016","#FeelTheBern", "#Bernie will lose"),
39                   tweets=1000, oauth=credential, timeout=10, lang='en' )
40      filterStream(file.name="Tweets_Donald_Trump.json",
41                   track=c("#NeverTrump","#AntiTrump", "#PresidentTrump"),
42                   tweets=1000, oauth=credential, timeout=10, lang='en' )
43      filterStream(file.name="Tweets_Marco_Rubio.json",
44                   track=c("#marcorubio","#RubioForPresident", "Rubio will lose"),
45                   tweets=1000, oauth=credential, timeout=10, lang='en' )
46      filterStream(file.name="Tweets_Ted_Cruz.json",
47                   track=c("#TedCruz","Cruz will win", "Cruz will lose"),
48                   tweets=1000, oauth=credential, timeout=10, lang='en' )
49      filterStream(file.name="Tweets_John_Kasich.json",
50                   track=c("#JohnKasich","Kasich will win", "Kasich will lose"),
51                   tweets=1000, oauth=credential, timeout=10, lang='en' )
52  })
53
```

Step 2: Sentiment Analysis

In this step we performed sentiment analysis on each tweet. To do this we are generating a list of words for each tweet and comparing this list with list of positive and negative words and checking how many words are positive and how many are negative.

After this we are marking a tweet as a positive sentiment if the weightage of positive words is more than the weightage of negative words.

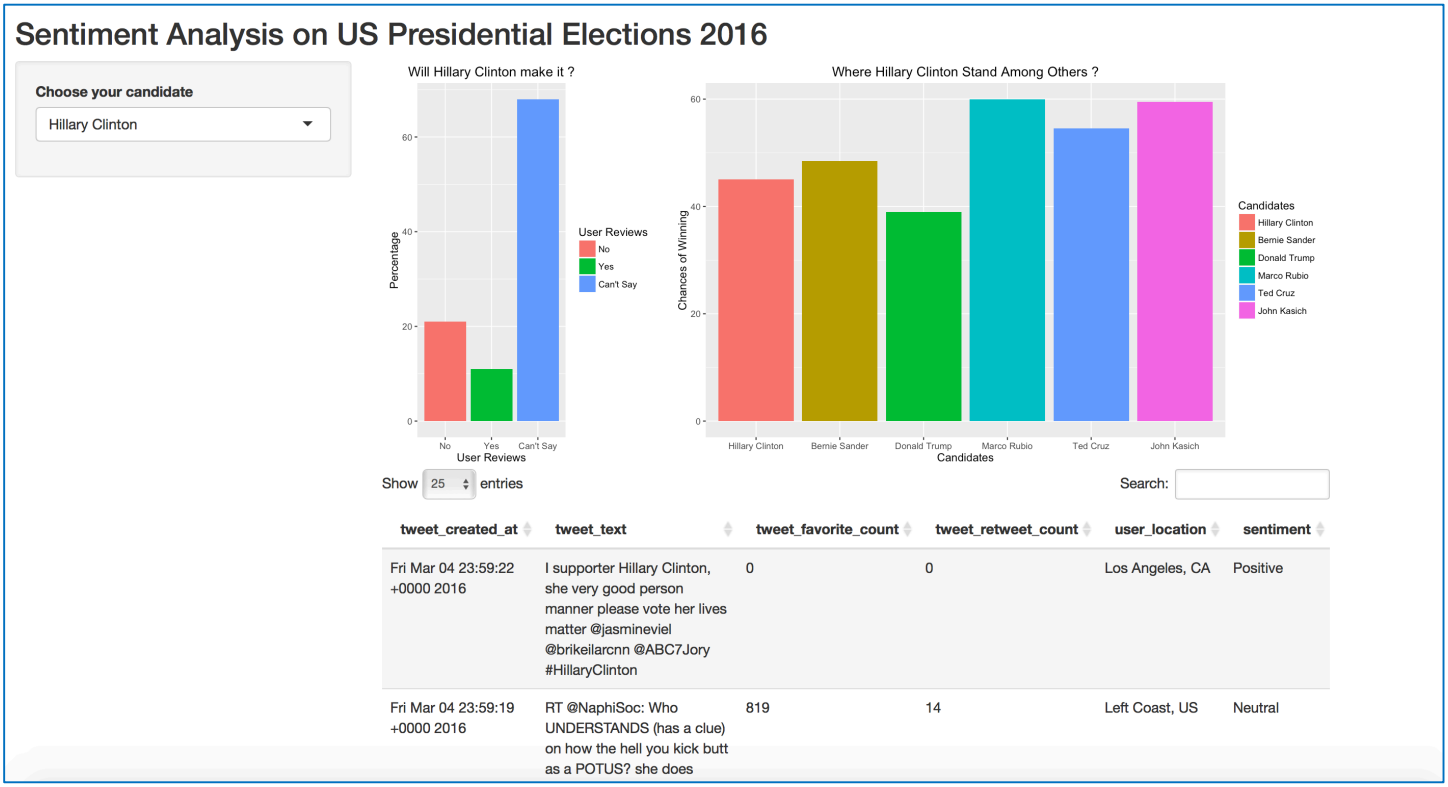
Since many a times when tweet contain both positive and negative words than generally its very difficult to predict what the user is saying. To combat this problem, we collected the popular +ve and –ve hashtags for each candidate and added those to the list of positive and negative words. This help us to get true picture of users sentiments.

```
    positive_matches = match(tweet_words_list, positive_words)
    negative_matches = match(tweet_words_list, negative_words)

    positive_matches = !is.na(positive_matches)
    negative_matches = !is.na(negative_matches)

    score = sum(positive_matches) - sum(negative_matches)

    if(score > 0){
      df_tweets[x, 13] <- "Positive"
      count_positive = count_positive +1
    } else if (score < 0){
      df_tweets[x, 13] <- "Negative"
      count_negative = count_negative +1
    } else {
      df_tweets[x, 13] <- "Neutral"
      count_neutral = count_neutral +1
    }

    x = x+1
    if (x > count_tweets){
      break
    }
  }

  df_tweets <- unique(df_tweets)

  df_sentiment <- data.frame(sentiment = "No", percent = round((count_negative*100)/count_tweets))
  df_sentiment <- rbind(df_sentiment, data.frame(sentiment = "Yes", percent = round((count_positive*100)/count_tweets)))
  df_sentiment <- rbind(df_sentiment, data.frame(sentiment = "Can't Say", percent = round((count_neutral*100)/count_tweets)


  output <- list("df_tweets" = df_tweets, "df_sentiment" = df_sentiment)
```

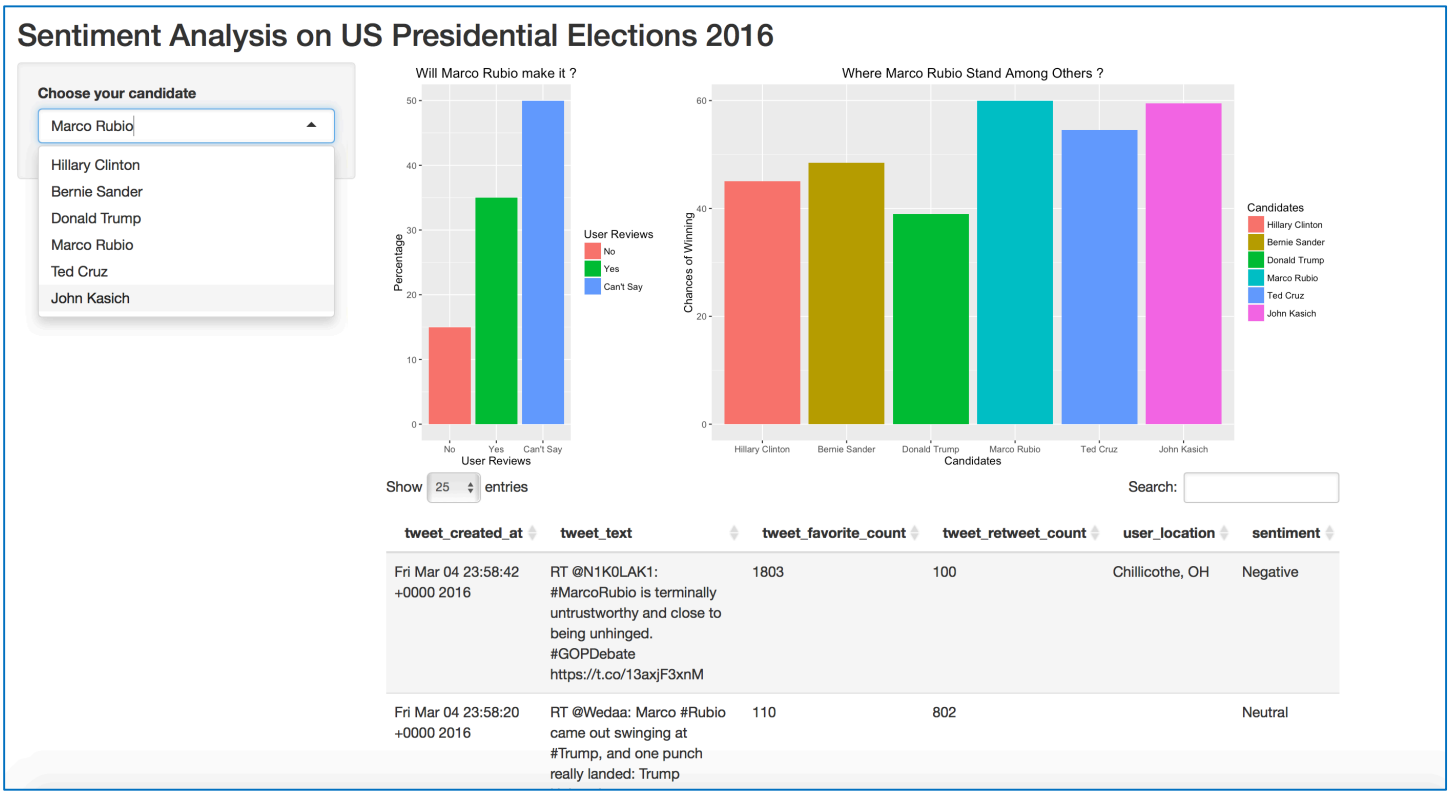
Step 3: Showing results on R Shiny

R Shiny portray analysis on very beautiful designed user interface, thus helps in conveying meaningful information

This is our UI for Shiny. We were able to predict that there is very tough competition among candidates. The graph on the right side shows the comparison among different candidates about their chances of winning based on users sentiments.



We can also choose our candidate and see their complete report on whether a particular candidate can win or not.



tweet_created_at	tweet_text	tweet_favorite_count	tweet_retweet_count	user_location	sentiment
Fri Mar 04 23:31:20 +0000 2016	Go home Donald you're drumpf - #TrumpRally #AntiTrump #drumpf https://t.co/NTQvLKaHrD	70	0		Negative
Fri Mar 04 23:28:09 +0000 2016	Linguistic study found that @realDonaldTrump speaks at a 4th grade level #trump #antitrump https://t.co/zuqC7msvNt https://t.co/uwl1Yk8Yym	93	0	Colorado, USA	Negative
Fri Mar 04 23:27:48 +0000 2016	STOP HATE #AntiTrump	1619	0	Sonora, México	Negative
Fri Mar 04 23:01:33 +0000 2016	@politico @Upworthy Children (our future), know more then 50% of GOP primary voters. https://t.co/9onDuUecil #ReactChannel #AntiTrump	388	0	Chicago, IL	Negative
Fri Mar 04 22:50:39 +0000 2016	RT @nashua40: Trump - June 16, 1980 Associated Press Billionaires Only - Americans can't afford to live here #AntiTrump https://t.co/MMbE5z…	0	1		Neutral
Fri Mar 04 22:48:37 +0000 2016	RT @move2canada2016: We can always move up to our neighbors...https://t.co/WP7Fd9jM27 … #antitrump #movingtocanada #trump https://t.co/h7aG…	50947	2	Ohio	Neutral

References and Acknowledgement:

I am grateful to the authors of these papers and their research on Sentiment Analysis. Their contribution has helped me deeply analyze user's sentiments.

http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html

Minqing Hu and Bing Liu. "Mining and Summarizing Customer Reviews." Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2004), Aug 22-25, 2004, Seattle, Washington, USA,

Bing Liu, Minqing Hu and Junsheng Cheng. "Opinion Observer: Analyzing and Comparing Opinions on the Web." Proceedings of the 14th International World Wide Web conference (WWW-2005), May 10-14, 2005, Chiba, Japan.