Capstone Project: Twitter Sentiment Analysis for Covid

Sandeep Kumar

08/03/2022

Introduction

What is Twitter Sentiment Analysis?

Twitter is an online micro blogging and social networking platform that enables users to send and read short character messages called “Tweets”.Registered users can read and post tweets,but those who are unregistered can only read them.

Hence Twitter is a public platform with a mine of public opionion of people all over the world and of all age categories.

Twitter Sentiment Analysis is the process of determining the emotional tone behind a series of words,used to gain an understanding of the attitudes, opionions and emotions expressed within an online mention.

Why Twitter Sentiment Analysis?

The applications for sentiment analysis are endless.It is extremely useful ib social media monitoring as it allows us to gain an overview of the wider public opinion behind certain topics however,it is also practical for use in business analystics and situations in which text needs to be analysed.

Sentiment analysis is in demand because of its efficiency.Thousands of text documents can be processed for sentiment in seconds,compared to hours it would take a team of people to manually complete.Because it is so efficient that many businesses are adopting text and sentiment analysis and incorporating it into their processes.

Overview

Tweets are imported using R and the data is cleaned by removing emoticons and URL’s etc.Lexical Analysis as well as Naive Bayes Classifier is used to predict the sentiment of tweets and subsequently express the opinion graphically through ggplots,histogram,wordcloud,tables etc.

Modeling Approach

1. Extraction of tweets

(i) Use rtweet package - provides an interface to the Twitter REST API
(ii) download the Tweets from Twitter using rtweet package.During authentication,we are redirected to a URL automatically where we click on Authorize app to download the data.

```
#loading the library
library(rtweet)
library(readr)
library(plyr)
library(stringr)
library(ggplot2)
library(wordcloud)
library(tm)
library(kableExtra)

# Setting current working directory
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))

#Search Twitter for 1000 tweets
#twitter.data <- search_tweets("Flipkart",n=1000,lang="en", include_rts=FALSE)

# Write CSV file to dir
#write_csv(twitter.data,"data.csv")

#Load tweets data from existing data file.
twitter.data.df <- read_csv("data.csv")
```

2. Cleaning of tweets

The tweets are cleaned by removing: 1. Extra punctuation 2. stopwords 3. URL's 4. &amp

```
tweets.df <- as.data.frame(twitter.data.df$text)
colnames(tweets.df)[1]<-"text"
tweets.df <- sapply(tweets.df$text, function(x) iconv(x, to='UTF-8-MAC', sub='byte'))
tweets.df <- gsub("@\\w+", "", tweets.df)
tweets.df <- gsub("#\\w+", '', tweets.df)
tweets.df <- gsub("RT\\w+", "", tweets.df)
tweets.df <- gsub("http.*", "", tweets.df)
tweets.df <- gsub("RT", "", tweets.df)
tweets.df <- sub("([.-])|[[:punct:]]", "\\1", tweets.df)
tweets.df <- sub("(['])|[[:punct:]]", "\\1", tweets.df)
tweets.df <- gsub("&amp", "", tweets.df)
```

3. Loading Words database

A database containing positive and negative words,is loaded into R.This is used for Lexical Analysis,where rhe words in the tweets are compared with the words in the database and the sentiment is predicted.

```
#Reading the Lexicon positive and negative words
pos <- readLines("positive_words.txt")
neg <- readLines("negative_words.txt")
```

4. Algorithms Used

Lexical Analysis: By comparing uni-grams to the pre-loaded word database,the tweet is assigned sentiment score -positive,negative or netural and overall score of corpus is calculated.

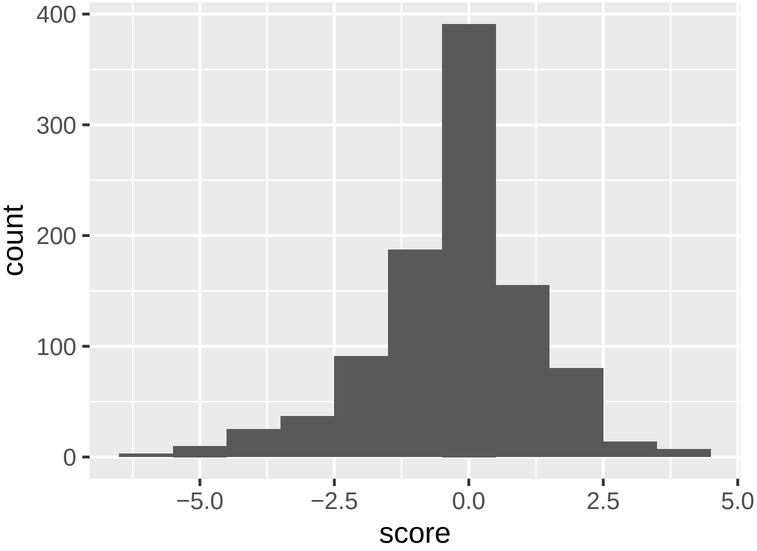
```r
#function to calculate sentiment score
score.sentiment <- function(sentences, pos.words, neg.words, .progress='none')
{
  # create simple array of scores with laply
  scores <- laply(sentences,
                 function(sentence, pos.words, neg.words)
                 {
                   # remove punctuation
                   sentence <- gsub("[[:punct:]]", "", sentence)
                   # remove control characters
                   sentence <- gsub("[[:cntrl:]]", "", sentence)
                   # remove digits
                   sentence <- gsub('\\d+', '', sentence)
                   # remove &amp
                   sentence <- gsub("&amp","",sentence)
                   #convert to lower
                   sentence <- tolower(sentence)

                   # split sentence into words with str_split (stringr package)
                   word.list <- str_split(sentence, "\\s+")
                   words <- unlist(word.list)

                   # compare words to the dictionaries of positive & negative terms
                   pos.matches <- match(words, pos)
                   neg.matches <- match(words, neg)

                   # get the position of the matched term or NA
                   # we just want a TRUE/FALSE
                   pos.matches <- !is.na(pos.matches)
                   neg.matches <- !is.na(neg.matches)

                   # final score
                   score <- sum(pos.matches) - sum(neg.matches)
                   return(score)
                 }, pos.words, neg.words, .progress=.progress )
  # data frame with scores for each sentence
  scores.df <- data.frame(text=sentences, score=scores)
  return(scores.df)
}
#sentiment score
scores_twitter <- score.sentiment(tweets.df, pos, neg, .progress='text')
```

```
##   |                                                                               |
```

5. Word Cloud

Histograms of positve,negative and overall score are analysing the intensity of emotion of the public using the tweets.

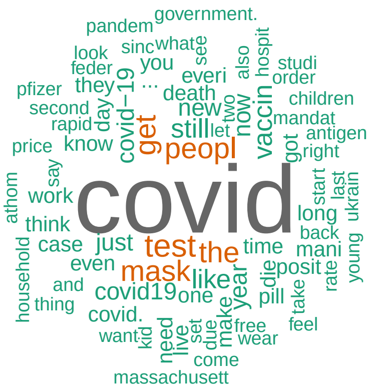
```
ggplot(scores_twitter, aes(x=score)) +geom_histogram(binwidth=1)
```



6. Word Cloud

A word cloud is a visual representation of the text data,typically used to depict keyword metadata on websited,or to visualize free form text.This is useful for quickly perceiving the most prominent terms and for locating a term alphabetically to determine its relative prominence. we have used the “tm” and “wordcloud” package.

```
# Create a Corpus
tweet_corpus <- Corpus(VectorSource(tweets.df))

# Remove stopwords
tweet_corpus <- tm_map(tweet_corpus,removeWords,stopwords("english"))
# Custom stopwords as a character vector
tweet_corpus <- tm_map(tweet_corpus,removeWords,c("the","will","can"))

# Text Stemming which reduces words to their root form
tweet_corpus <- tm_map(tweet_corpus,stemDocument)

tweet_corpus <- TermDocumentMatrix(tweet_corpus)
matrix <- as.matrix(tweet_corpus)
words <- sort(rowSums(matrix),decreasing = TRUE)
tweet_corpus <- data.frame(word = names(words),freq = words)
# Generating wordcloud
wordcloud(words = tweet_corpus$word,freq = tweet_corpus$freq,min.freq = 10,max.words = 80,random.order =
```



Results/Conclusion

Below is the sentiment analysis of the tweets related to hastag “covid”.

```
summary(scores_twitter) %>% knitr::kable(booktabs=T)
```

text	score
Length:1000	Min. :-6.000
Class :character	1st Qu.:-1.000
Mode :character	Median : 0.000
NA	Mean :-0.263
NA	3rd Qu.: 1.000
NA	Max. : 4.000

it is observed that public opinion towards the Covid of is balanced as peoples are now used to their daily life.So we can say now peoples are taking the covid19 casual ly.