Sentiment Analysis With R

Tainara Camila Zacarias

30/03/2021

Mini-Project - Sentiment Analysis of Twitter Data

Este projeto tem como objetivo capturar dados da rede social Twitter a respeito de um tópico específico e realizar análises de sentimento sobre esses dados obtidos.

Inicialment serão listados todos os pacotes necessários para a execução do projeto. Em seguida será descrito o processo de autenticação e extração dos dados do Twitter. Na sequência é apresentado os processos realizados para o pré processamento dos dados obtidos, além de uma breve análise dos mesmos. Por fim, serão feitas duas análises principais: o sentimento geral e as emoções envolvidas nos tweets extraídos.

Pacotes utilizados

```
#install.packages("rtweet")
#install.packages("tm")
#install.packages("SnowballC")
#install.packages("wordcloud")
#install.packages("RColorBrewer")
#install.packages("SentimentAnalysis")
#install.packages("syuzhet")
#install.packages("textreg")

library(syuzhet)
library(textreg)
library(tibble)
library(ggplot2)
library(SentimentAnalysis)
library(RColorBrewer)
library(wordcloud)
library(SnowballC)
library(tm)
library(rtweet)
library(dplyr)
source("utils.R")
```

Etapa 1 - Autenticação

Nessa etapa é realizada a autenticação das credenciais da API do twitter que será utilizada para este projeto. Esta etapa é necessária para que você consiga realizar a extração dos dados.

Para mais informações sobre como criar e usar uma Twitter API acesse: Obtaining and using access tokens

```
app_name = ""
consumer_key = ""
consumer_secret = ""
access_token = ""
access_secret = ""

token = create_token(app = app_name,
             consumer_key =  consumer_key,
             consumer_secret = consumer_secret,
             access_token = access_token,
             access_secret = access_secret)
```

Etapa 2 - Extração dos dados

Para a extração dos dados será utilizado o pacote rtweet. Neste projeto serão extraídos 1500 tweets que contenha a palavra "Bolsonaro". É importante ressaltar que quanto maior a quantidade de tweets a serem extraídos maior será o tempo de extração.

```
tweets <- search_tweets(
  "Bolsonaro",
  n = 1500,
  include_rts = FALSE,
  lang = "en",
  token = token)
```

Etapa 2.1 (Opcional) - Salvando os dados extraídos em um CSV

Salvar os dados obtidos em um CSV é interessante pois, garante que os dados estarão disponíveis em qualquer momento. Para armazenar esses dados eles devem ser inicialmente convertidos e então salvos em um csv.

```
# FYI: Some of the tweet's data frame columns are of type list. The data frame
# is no longer 2-dimensional and can't be exported to a 2d csv-file directly.
# It's necessary to format those columns before export. In here, the columns are
# being converted to char type.

tweets_converted <- data.frame(
  lapply(tweets, as.character),
  stringsAsFactors=FALSE)

write.csv(tweets_converted, 'BolsonaroTweetsEn.csv')
```

Etapa 3 - Tratamento dos dados coletados

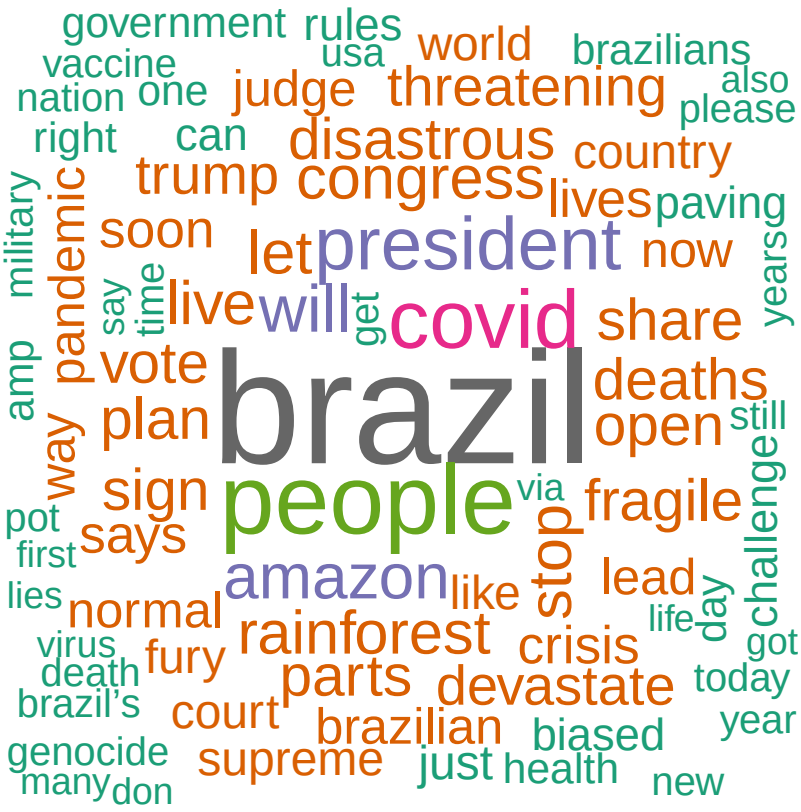
Nesta etapa os dados extraídos passam por um processo de limpeza e transformação. Primeiramente é extraído apenas os textos dos tweets extraídos. Em seguida esses dados passam por uma limpeza onde é removido as pontuações, links, usernames, números, passadas para low case e etc. Com os dados limpos, eles são convetidos em um objeto do tipo Corpus e as stopwords são removidas, juntamente das palavras "jair" "bolsonaro" e "lula" .

```r
# Get only tweets text
tweets_text <- tweets$text

# Cleaning
tweets_text_cleanned <- cleanTweets(tweets_text)

# Corpus
tweets_corpus <- VCorpus(VectorSource(tweets_text_cleanned))

# Remove stopwords
tweets_corpus <- tm_map(tweets_corpus, removeWords, stopwords('english'))

# Remove your own stop words
tweets_corpus <- tm_map(
  tweets_corpus,
  removeWords,
  c("jair", "bolsonaro", "lula"))
```

Etapa 4 - Wordcloud e palavras que aparecem com mais frequência

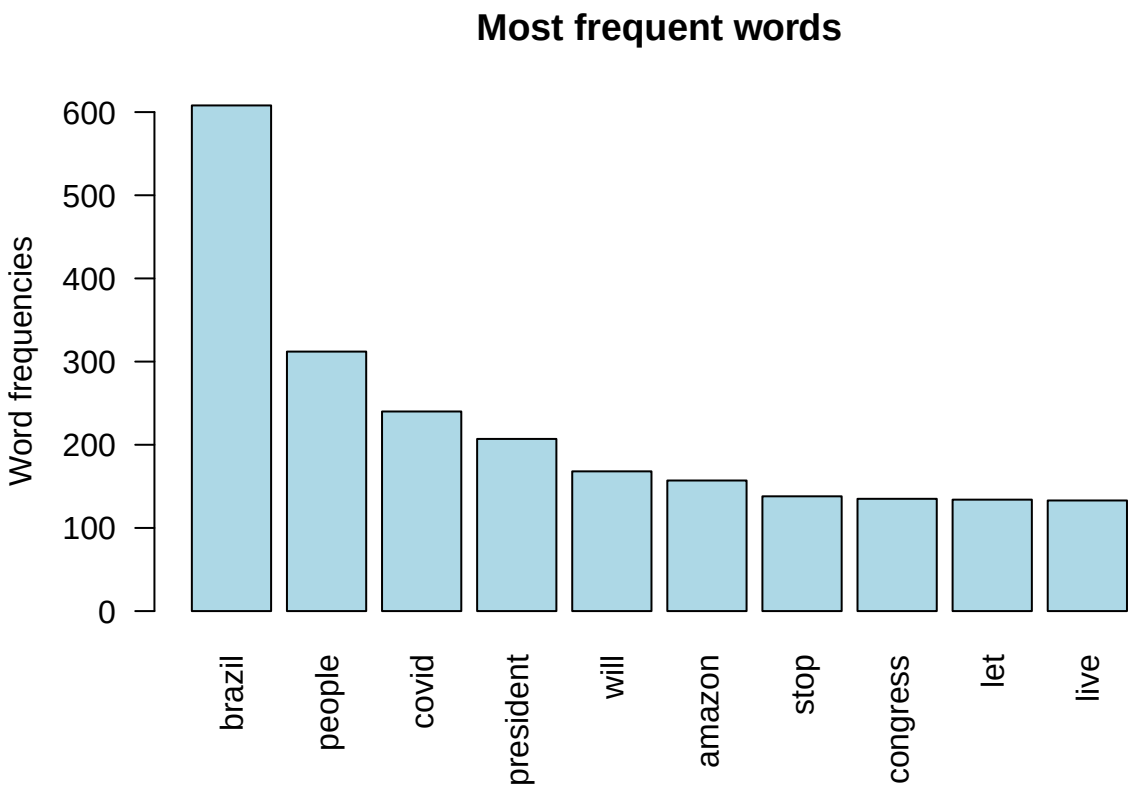
A nuvem de palavras (wordcloud) é utilizada para a visualização das palavras que aparecem com mais frequência e a relação entre elas.

```r
wordcloud(tweets_corpus,
          min.freq = 2,
          scale = c(5,1),
          max.word = 100,
          random.order = F,
          colors = brewer.pal(8,"Dark2"))
```



É interessante também conhecer as palavras de maior frequência de uma maneira mais sucinta, e pode-se fazer isso através da utilização da função `DocumentTermMatrix` do pacote tm.

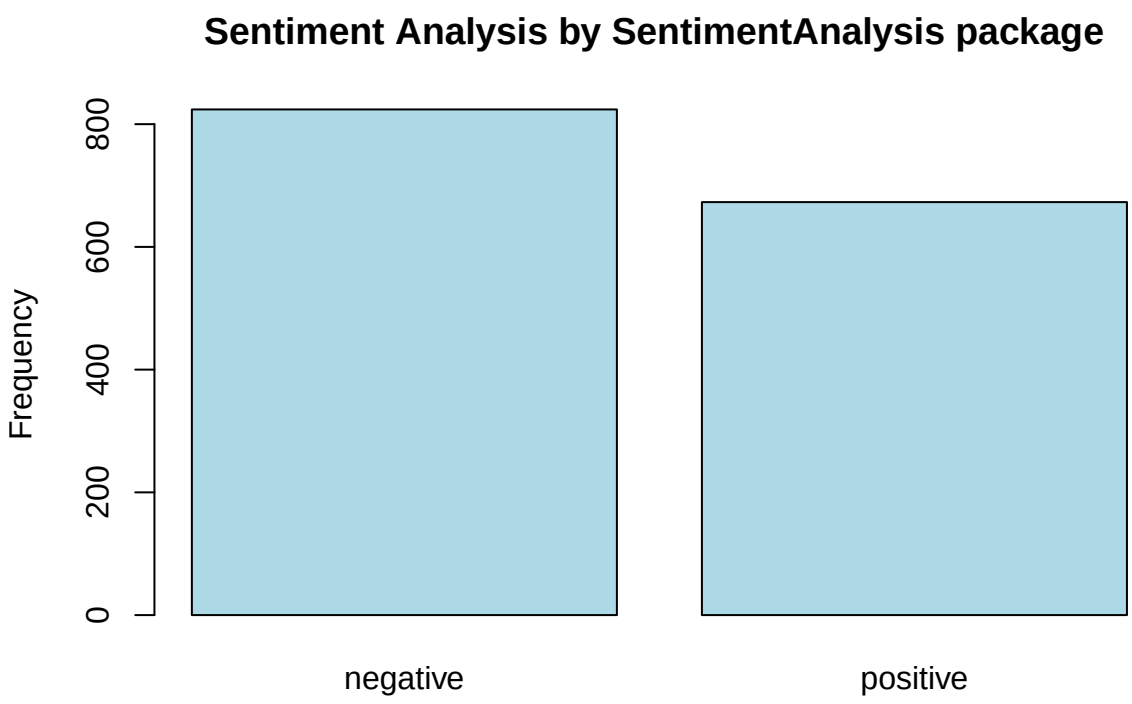
```
document_matrix <- as.matrix(TermDocumentMatrix(tweets_corpus))

words_frequency <- sort(rowSums(document_matrix), decreasing=TRUE)

words_frequency_df <- data.frame(
  word = names(words_frequency),
  freq = words_frequency)

barplot(words_frequency_df[1:10,]$freq,
        las = 2,
        names.arg = words_frequency_df[1:10,]$word,
        col ="lightblue", main ="Most frequent words",
        ylab = "Word frequencies")
```



Etapa 5 - Análise dos sentimentos

Através do pacote sentimentAnalysis é possível analizar o sentimento geral em torno dos tweets apresentados. Esse sentimento pode ser negativo ou posito.

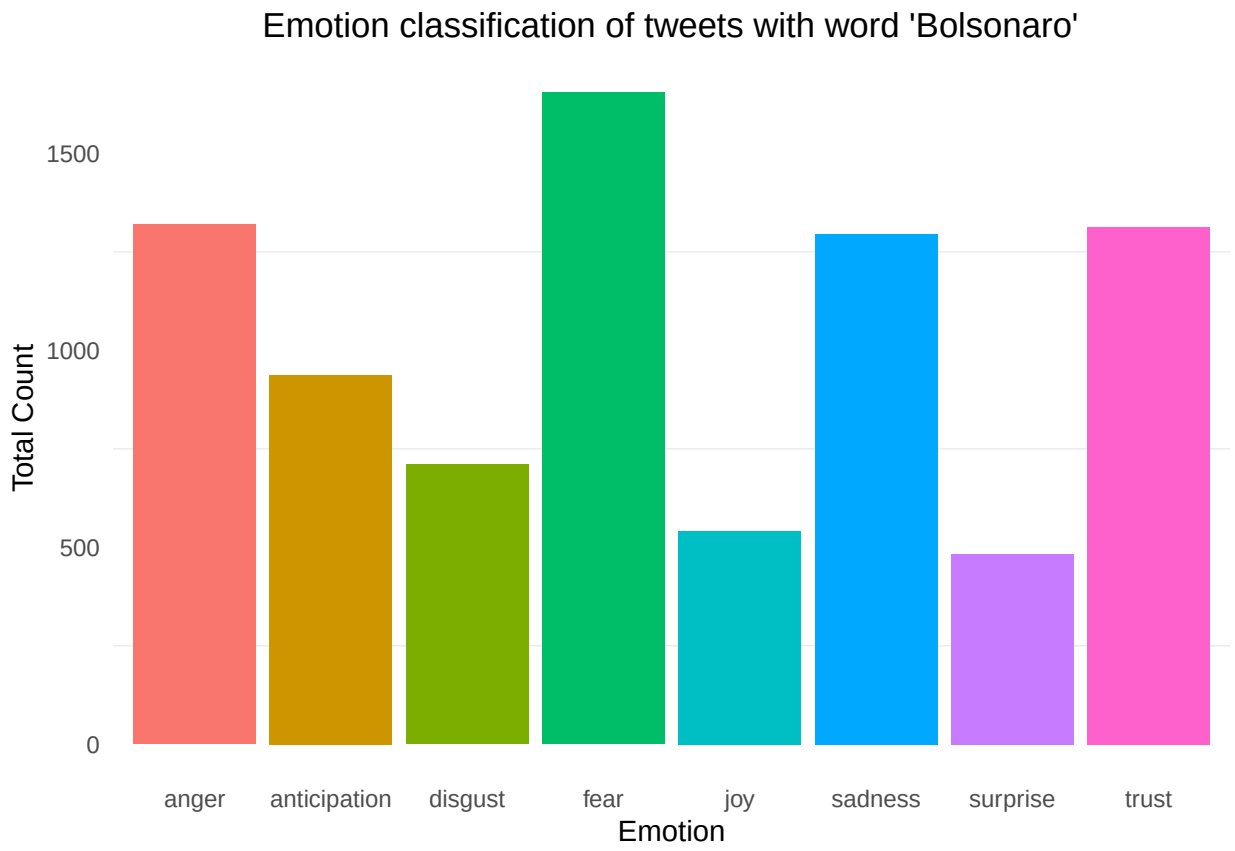
```
sentiment_analysis_v1_0 <- analyzeSentiment(tweets_corpus)

barplot(table(convertToBinaryResponse(sentiment_analysis_v1_0$SentimentLM)),
        col ="lightblue",
        main ="Sentiment Analysis by SentimentAnalysis package ",
        ylab = "Frequency")
```



No caso dos dados extraídos, pode-se observar que o sentimento geral em torno a palavra “Bolsonaro” é mais negativo que positivo. Somente com essa analíse os tomadores de decisão já podem ficar mais alertas e começar a pensar em medidas para que o sentimento em torno desse assunto seja mais positivo.

Pode-se analisar esses sentimento um pouco mais a fundo, através da classificação de emoções. A classificação das emoções pode ser obtida através do pacote Syuzhet

```
# Transform corpus into char
tweets_text_pre_processed <- convert.tm.to.character(tweets_corpus)

# Get emotions
sentiment_analysis_v2_0 <- get_nrc_sentiment(tweets_text_pre_processed)

# Plot
sentiment_analysis_v2_df <- as.data.frame(colSums(sentiment_analysis_v2_0[1:8]))
sentiment_analysis_v2_df <- rownames_to_column(sentiment_analysis_v2_df)
colnames(sentiment_analysis_v2_df) <- c("emotion", "count")

ggplot(sentiment_analysis_v2_df, aes(x = emotion, y = count, fill = emotion)) +
  geom_bar(stat = "identity") +
  theme_minimal() +
  theme(legend.position="none", panel.grid.major = element_blank()) +
  labs( x = "Emotion", y = "Total Count") +
  ggtitle("Emotion classification of tweets with word 'Bolsonaro'") +
  theme(plot.title = element_text(hjust=0.5))
```


Emotion classification of tweets with word 'Bolsonaro'
Total Count
1500
1000
500
0
anger
anticipation
disgust
fear
joy
sadness
surprise
trust
Emotion