

Analise de Sentimentos - Rio2016

Guilherme Righetto

12 de fevereiro de 2017

Análise de Sentimentos

Este é um projeto sobre a análise de sentimentos a partir dos tweets das olimpíadas do Rio 2016. O objetivo é capturar dados do Twitter e realizar a criação de uma nuvem das palavras e de um análise de sentimentos (Positivo/Negativo/Neutro). Todo o projeto será descrito nas próximas seções.

Pacotes

```
library(twitterR)
library(tm)
library(wordcloud)
library(SnowballC)
library(Rstem)
library(sentiment)
library(plotly)
library(plyr)
```

Autenticação do Twitter

O código apresentado realiza a autenticação realizada pelo pacote 'twitterR', é válido ressaltar que é necessário criar um aplicação na área de desenvolvedor do twitter.

```
api_key <- 'IpbVirlaHHLkTP8MfkL5Yuavm'
api_secret <- '7f0U01fQj5QBLNxRxo85k9k1Hze7HDl0MtaOBjOKTwynnSCula'
token <- '830527057423691776-hVh1j10t8yNsX6f98NVwyEAfwFvYts0'
token_secret <- 'vAwDfU9buhjd7yaFrdiv5zcCo2HWefTd6amN7L5K79LVY'

setup_twitter_oauth(api_key, api_secret, token, token_secret)
```

```
## [1] "Using direct authentication"
```

Carregar Tweets

As próximas funções executam uma busca de 500 tweets com a string de busca '#Rio2016' em português e depois transformam os dados em um data frame.

```
tweets <- searchTwitter("#Rio2016",n=500, lang = 'pt')
tweetpt <- twListToDF(tweets)
```

Limpeza e Pré-processamento dos Dados

Nesta etapa, será realizado a limpeza de dados como também o agrupamento de todos os tweets utilizando o pacote de text mining(tm). A limpeza realizada é referente aos caracteres especiais, acentuação, números e stopwords.


```

tweetst = gsub("(RT|via)((?:\\b\\W*@\\w+)+)", "", tweetst)
tweetst = gsub("@\\w+", "", tweetst)
tweetst = gsub("[:punct:]", "", tweetst)
tweetst = gsub("[:digit:]", "", tweetst)
tweetst = gsub("http\\w+", "", tweetst)
tweetst = gsub("https://.+","",tweetst)
tweetst = gsub("[ \\t]{2,}", "", tweetst)
tweetst = gsub("^\\s+|\\s+$", "", tweetst)
tweetst = gsub("<.+>", "",tweetst)
tweetst = tolower(tweetst)
tweetst = tweetst[!is.na(tweetst)]
names(tweetst) = NULL

```

Classificação

O código a seguir utiliza a função ‘classify_polarity’ que classifica os tweets em três classes (Positive, Negative e Neutral) utilizando o algoritmo Naive Bayes.

```

class_pol = classify_polarity(tweetst, algorithm = "bayes")
polarity = class_pol[,4]

```

Adequação

Nesta etapa será realizado uma adequação do retorno do classificador para construir um gráfico de pizza com melhor entendimento.

```

sent_df = data.frame(text = tweetst, polarity = polarity, stringsAsFactors = FALSE)

sent_df$polarity = gsub("positive", "POSITIVO", sent_df$polarity)
sent_df$polarity = gsub("negative", "NEGATIVO", sent_df$polarity)
sent_df$polarity = gsub("neutral", "NEUTRO", sent_df$polarity)

sent_df <- count(sent_df, ~polarity)
sent_df$polarity <- paste(sent_df$polarity, sent_df$freq)

```

Gráfico de Pizza

Construção do gráfico de pizza utilizando o pacote ‘plotly’.

```

p <- plot_ly(sent_df, labels = ~polarity, values = ~freq, type = 'pie') %>%
  layout(title = 'Análise de sentimentos sobre as Olimpíadas Rio2016',
    xaxis = list(showgrid = FALSE, zeroline = FALSE, showticklabels = FALSE),
    yaxis = list(showgrid = FALSE, zeroline = FALSE, showticklabels = FALSE))

```

Análise de sentimentos sobre as Olimpíadas Rio2016

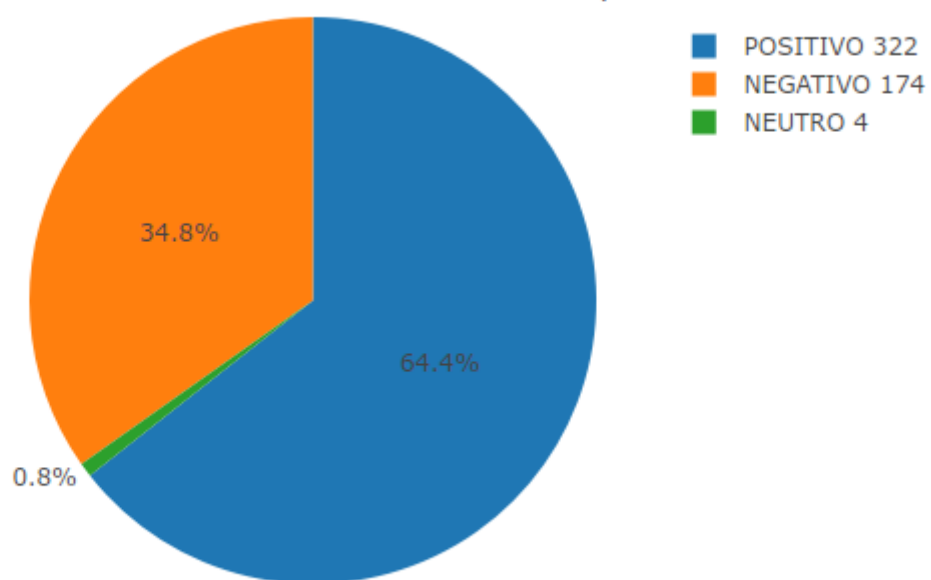


Figure 1: Análise de Sentimentos.