My first Twitter data scrape

Rodrigo Esteves de Lima-Lopes
State University of Campinas
rll307@unicamp.br

Contents

1 Introduction

Our main objective here is to have first contact with network with a data scraping package. In this case, `rtweet`. Twitter has been one of the oldest surviving social media, and it also has been an important source for data studying at the Computer Mediated Communication studies in the last few years. But please, take into account that the data we collect might be influenced by a number of factors:

1. My location: Twitter's algorithm is known to change the results depending of our location
2. A different kind of account (professional, personal or premium) offers different results in terms of how many of twitters we my get
3. Our network capacity might influence the results

1.1 What are we going to need

1. A valid Twitter account
2. The package rtweet

1.1.1 Responsible data use

Please, keep in mind that any data scraping should be done in accordance to Twitter's terms and conditions. # Scraping some data

1.2 Twitter locations

First we are going to get some insights on what is trending in our location. So we start by checking which are the locations available:

```
my.trends<-trends_available()
```

If we have a close look, `my.trends<-trends_available()` delivers a table with numbers, cities and countries. Please, mind you that some locations might not be listed. In this case, I would choose your countries location. I am from São Paulo - Brazil, so I will try to get the trends available there. If we look at the table, São Paulo is ID `455827`. So we will get the trends using this number.

```
SP.trends<-get_trends(woeid=455827)
```

```
head(SP.trends)
```

```
##                trend                                                url
## 1             gremio               http://twitter.com/search?q=gremio
## 2     Bruno Henrique http://twitter.com/search?q=%22Bruno+Henrique%22
## 3           Flamengo             http://twitter.com/search?q=Flamengo
## 4            Vitinho              http://twitter.com/search?q=Vitinho
## 5 #provadofazendeiro http://twitter.com/search?q=%23provadofazendeiro
## 6              Lisca                http://twitter.com/search?q=Lisca
##   promoted_content                query tweet_volume     place  woeid
## 1               NA               gremio        67817 São Paulo 455827
## 2               NA %22Bruno+Henrique%22        14722 São Paulo 455827
## 3               NA             Flamengo       586196 São Paulo 455827
## 4               NA              Vitinho       214791 São Paulo 455827
## 5               NA %23provadofazendeiro       131542 São Paulo 455827
## 6               NA                Lisca        23419 São Paulo 455827
##                  as_of          created_at
## 1 2020-11-19 09:55:52 2020-11-18 14:23:56
## 2 2020-11-19 09:55:52 2020-11-18 14:23:56
## 3 2020-11-19 09:55:52 2020-11-18 14:23:56
## 4 2020-11-19 09:55:52 2020-11-18 14:23:56
## 5 2020-11-19 09:55:52 2020-11-18 14:23:56
## 6 2020-11-19 09:55:52 2020-11-18 14:23:56
```

Again we have a table. But please, mind you it is a snapshot of Twitter at the moment data was collected, it tends to change, sometimes, by the minute.

1.3 Getting some tweets

In my data, the hashtag `#VerdadeDosFatos` called my attention, so I will search for it. There are two ways to do so:

1. `stream_tweets()`: searches tweets for a given period of time
2. `search_tweets()`: searches tweets until it gets specified number of occurrences

1.3.1 stream_tweets

- **Advantages**: it collects as much tweets as possible in a given period of time
- **Disadvantages**: it tends to get connection and parsing errors when we search for long periods of time. There is a function called recover_stream.R, written by Johannes Gruber (who we are quite deeply thankful) and available here, that might sort the problem most sometimes. But *most* means: if our file is too much damaged, it will not work as we intend.

Let us make some search using `stream_tweets`:

```
SP.T1 <- stream_tweets('#VerdadeDosFatos',
                       timeout = 60, #in seconds
                       file_name='verdade_01', # it saves a file, not a variable
                       parse=TRUE)
```

Now to load this tweets we will need the following command:

```
SP.tweets <- parse_stream("verdade_01.json")
```

If we look at this file, there is a lot of possible variables to explore, over 90 columns with a lot of information regarding our tweets.

1.3.2 search_tweets()

- **Advantages**: it collects a certain number of tweets. Always returns nice parsed files.
- **Disadvantages**: if you do not have a researcher or premium account, number of instances might be limited.

Due to time, we will search for some tweets only:

```
SP.tweets2 <- search_tweets(
  "#VerdadeDosFatos", n = 1000, include_rts = TRUE
  )
```

The result is a similar data frame. However, I have posted now and in the past (backwards) tweets.

1.4 Some bonus features

Let us get the timeline form a politician:

```
boulos <- get_timeline("GuilhermeBoulos",n=1000)
```

Let us get his followers

```
boulos.flw <- get_followers("GuilhermeBoulos", n = 75000)
```

Now let us get some information regarding some of those followers

```
boulos.flw2 <- boulos.flw[1:100,]
info <- lookup_users(boulos.flw2$user_id)
```

Let us get some users with our hashtag in their bios:

```
users <- search_users("#VerdadeDosFatos", n = 1000)
```

Let us get the timelines for the two candidates in the second round of São Paulos's elections:

```
prefeitura <- get_timelines(c("brunocovas", "GuilhermeBoulos"), n = 3200)
```

Now let us plot the frequency:

```
library(ggplot2)

prefeitura %>%
  dplyr::filter(created_at > "2020-08-01") %>%
  dplyr::group_by(screen_name) %>%
  ts_plot("days", trim = 7L) +
  ggplot2::geom_point() +
  ggplot2::theme_minimal() +
  ggplot2::theme(
    legend.title = ggplot2::element_blank(),
    legend.position = "bottom",
    plot.title = ggplot2::element_text(face = "bold")) +
  ggplot2::labs(
    x = NULL, y = NULL,
    title = "Frequency of Twitter statuses posted by Candidates in São Paulo",
    subtitle = "Twitter status (tweet) counts aggregated by day from August 2020",
    caption = "\nSource: Data collected from Twitter's REST API via rtweet"
  )
```

Which conclusions can we take?

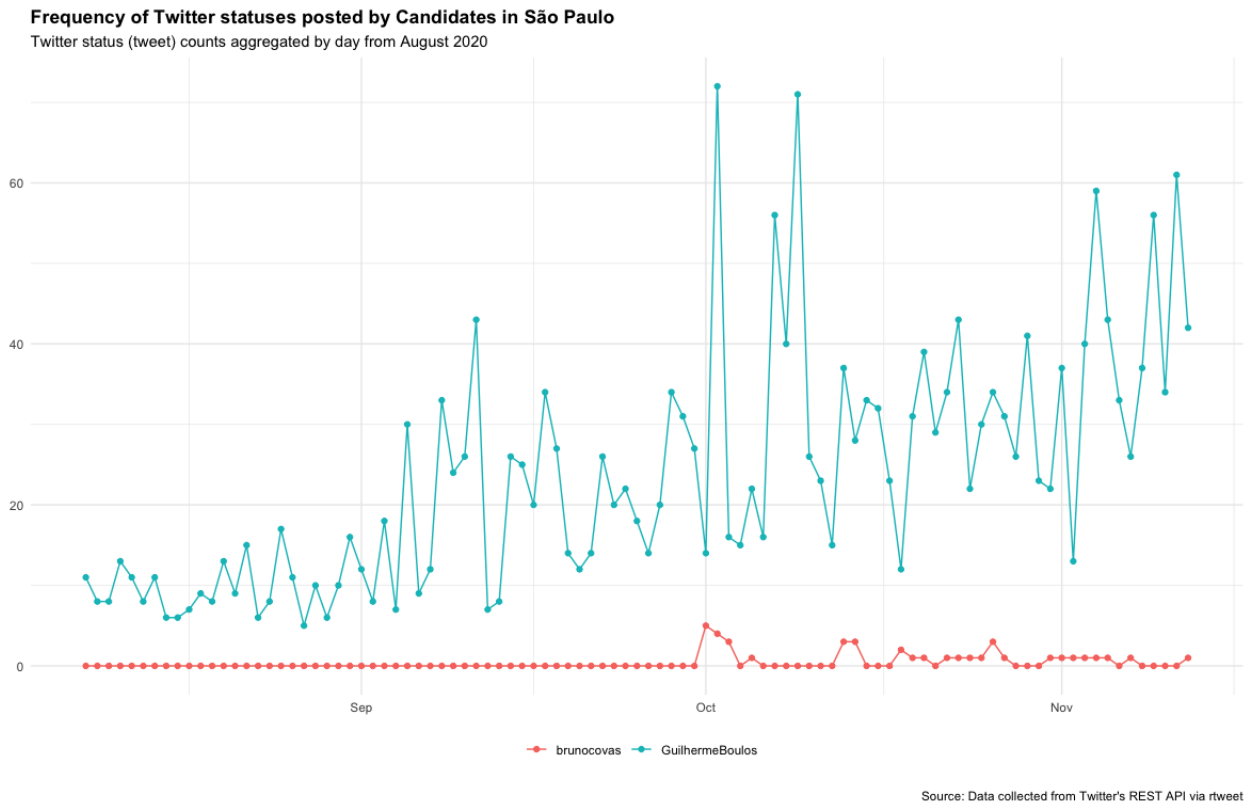

Figure 1: Tweets Comparison