Magí Izquierdo Lechuga

Tweetmeroteca project report

Contents

Summary

Tweetmeroteca, is an APP based on the live feeding of the social media platform called Twitter. On Twitter, everything is visually designed for the customer/user with a variety of colours, icons and links to others users, feeds, and comments. In the Shiny APP developed, this style tries to stay alive, while the focus of the information retrieved is intended for an analysis on how the community is reacting to a certain Hashtag (Descriptive word attached to the Tweet by the user). It is giving a quick insight on what the reaction of the community is and where is it happening in a summarized view with no more than 4 tabs of information. This APP can be useful for companies that would like to have a greater impact in certain regions, social community or matter of study having a previsualization on where, when and how is the people sharing or reacting to a specific Hashtag. Additionally, online deployment is adapted to mobile devices.

Background Research

Originally, this project has been developed as a normal shiny app with the intention of applying a CSS style. However, after reviewing on some options and other modules of the subject it came the decision on create a Dashboard. Something similar on what *gadenbuie* from the github community created for a 2019 conference [1]. This app generates a friendly user dashboard with clear information and an easy to navigate format with a left side menu with different tabs to select avoiding the confusion of more upper tabs that can be sometimes interpreted as browser tabs. In order to create the dashboard, the package “shinydashboard” was found to be very useful and easy to deploy in the R environment. However, the most relevant information about shiny dashboard design was found on another rstudio github webpage [2].

A different point of interest that, as well required from some research, was the application of a map with location indicators. It is not frequently found to have access on where the Tweets have been created. Therefore, the most visited source of information has been *r-spatial*, a website which perfectly explains how to add a map into Rstudio using “ggplot2” and the “sf” library [3]. However, other websites have been visited to understand the logic behind Google API´s and how to use it in the R environment [4],[5],[6].

Additional research has been done on API´s implementation and Tokens creation. However, Twitter is not able to be automatically authorized when loading the app to shinyapps.io and the reactive tale cannot be loaded but following community indications a workaround has been found [9].

Data Source

The principal source of data is the social media platform called Twitter. However, in order to retrieve the information without the use of a token, as Twitter has not provided access to it, has been an article from *Sharon Machlis* in the Infoworld website [7]. This article allows the reader to create a table of the main Twitter points of interests and have the information in a filterable and sort reactive table. Therefore, the rest of the app and the project has been based in this table adding new items to be retrieved from the website and making it reactive to other inputs in a nice shiny dashboard page. It allows, as well, to retrieve any link attached to the Tweet in a clickable URL format. Twitter developer's website allows a longer list of items to be selected but for the purpose of this project only the location of the Tweet has been considered to be added. Additionally, the location data has been transformed into latitude and longitude using the "geocode()" function which thanks to Google API can read the location, transform it into latitude and longitude and then can be applied into a coordinates map.

Criticism

This project has been an overall challenge but I have enjoyed greatly the overcoming of the problems during the research. Several webpages and sources of information has been read until finding the good one or the ones that mixing information helps you to succeed.

The main challenges faced have been:

· Twitter API and Token: It has not been possible to gain access to Twitter Token. However, finding a workaround of accessing the API and retrieve the information has been of big help to continue with the project.

· Animated plot: All the code used in class has not been applied as the reactive table didn´t allowed to do so. However, it was interesting finding the code to create an animated gif that can be automatically played when opened in the desired destination.

· Map plot: It is a plot that has not been worked before in class, only briefly referenced in some examples and I wanted to create one. All resources found were referencing the "leaflet()" function but it was not able to read the location. Google API and "geocode()" function have been a hard bone but seeing the results it has been worth it of the effort.

· Frequency ggplot: ggplot is a great plotting tool however when trying to plot a count, cumulative sum, etc. It does not allow it unless it is calculated beforehand. After finally deciding on dismiss the plot the ggplot option "stat_bin()" presented the opportunity to create a histogram of the count of Tweets per date. A complete success after all the time invested in having the frequency of Tweets per date.

· Shinyapp.io deployment: The deployment into the website has been only reading the dashboard and the input information. However, it has not deployed correctly the code used to retrieve the Tweet table and therefore all the code based in the table is not visible in the

website. Nevertheless, after trying several solutions offered by the community it has finally successfully worked.

Tweetmeroteca can be highlighted for:

· Main Dashboard page, it provides an overview of the Hashtag to have a quick description of it and decide if it is worth of additional analysis.

· Time lines, the analysis page provides two interesting plots based on a timeline of the Hashtag, one for likes and another one for frequency of tweets.

· Geo map, this map is a visual summary of all the locations where the hashtag has been used without the need of complicated tables or the limited information of just the most frequent location. You can discover the use of the Hashtag based on the cumulative points per region on the map.

· Style, keeping a fresh, simple and opened style thanks to the dashboard structure which allows to hide and unhide the menu and with the brand colours and logo of the company as part of the background.

On the other hand, the main weaknesses are:

· Loading times: It is maybe the biggest of the weaknesses as the number of tweets selected on the slider has been reduced to a maximum of 1000. If the selector is kept at 100 instances the animated plot rendering and the map loading time can take up to 20 seconds.

· Outcome based on a reactive table: Having all the code depending on a reactive table can lead to an unsuccessful result if the table is not properly loaded. It makes the rest of the code hard to manipulate with conditions to an uncertain data.

Link

https://kawkawbala.shinyapps.io/VIZPROJECT/

Twitter Sign in credentials (if requested):

User Name: Tweetmeroteca
Password: Tweetmeroteca01

History

05/04/2020

How to Access Twitter API and generate a table [7]

06/04/2020

How to transform to dashboard [2].
Add location (GEO) to the reactive table looking at the structure of the code.

11/04/2020

Extract values from reactive table focus on GEO to create a map and include location of tweets

12/04/2020

Try of different models and ways to plot world map and points. Discovered how to add map in tab 4 [4],[5],[6].
Discovered how to extract info from reactive table using empty" () "at the end when calling the table.
Reactive info does not have required format to be plotted on the map.
Find function "geocode" which transforms names into long and lat. Need to create Google API.
Create Google API
(Again) Reactive info does not have required format to be plotted on the map.
Find a work around changing the type of map plotted [3].

13/04/2020

Generated the info Boxes and text of the Dashboard.
Created the Twitter Background and changed opacity to allow text to be visible.
After 12h research - Created the Animated Gif [8].

17/04/2020

Created plot for the frequency of tweets among time using "stat_bin()" in ggplot.

18/04/2020

App deployed to Shinyapp.io, but Tweet content is not loaded. Get the error:
"> rsconnect::showLogs()
Error in generateAppName(appTitle, appPath, account, unique = FALSE) :
The generated app name '3' is invalid. Include at least 3 alphanumeric characters in the title."
Visited all kind of blogs and tested several solutions (change directory, deploy app manually, copy code and create a new shiny app in a different folder, …)

19/04/20

Finally, found a solution for the API credentials and Token to be accepted on shinyapps.io [9].
Submission of the project.

Reference

[1] GitHub. 2019. *Gadenbuie/Tweet-Conf-Dash*. [online] Available at: <https://github.com/gadenbuie/tweet-conf-dash> [Accessed 1 April 2020].

[2] Rstudio.github.io. 2014. *Shiny Dashboard Structure*. [online] Available at: <https://rstudio.github.io/shinydashboard/structure.html> [Accessed 6 April 2020].

[3] Basille, M., 2018. *Drawing Beautiful Maps Programmatically With R, Sf And Ggplot2 — Part 1: Basics*. [online] R-spatial.org. Available at: <https://www.r-spatial.org/r/2018/10/25/ggplot2-sf.html> [Accessed 12 April 2020].

[4] Datanovia. 2020. *How To Create A Map Using Ggplot2: The Best Reference - Datanovia*. [online] Available at: <https://www.datanovia.com/en/blog/how-to-create-a-map-using-ggplot2/> [Accessed 12 April 2020].

[5] Stack Overflow. 2015. *How Do I Plot US Cities Using Ggplot?*. [online] Available at: <https://stackoverflow.com/questions/29037851/how-do-i-plot-us-cities-using-ggplot> [Accessed 12 April 2020].

[6] Prevos, P., 2018. *Geocoding With Ggmap And The Google API – R-Craft*. [online] R-Craft. Available at: <https://www.r-craft.org/r-news/geocoding-with-ggmap-and-the-google-api/> [Accessed 12 April 2020].

[7] Machlis, S., 2020. *Create A Shiny App To Search Twitter With Rtweet And R*. [online] InfoWorld. Available at: <https://www.infoworld.com/article/3516150/create-a-shiny-app-to-search-twitter-with-rtweet-and-r.html> [Accessed 5 April 2020].

[8] Stack Overflow. 2016. *How To Create And Display An Animated GIF In Shiny?*. [online] Available at: <https://stackoverflow.com/questions/35421923/how-to-create-and-display-an-animated-gif-in-shiny> [Accessed 13 April 2020].

[9] GitHub. 2019. *Authentication Problem · Issue #202 · Ropensci/Rtweet*. [online] Available at: <https://github.com/ropensci/rtweet/issues/202> [Accessed 19 April 2020].