Data 607 Week 9 Homework

Jean Jimenez

2023-10-24

Introduction

This week, we learned about web APIs. Web APIs are cool because they allow for an interface for developers to use to easily retrieve the data that they need. For this weeks homework assignment, we had to go through the New York Times' API. We had to chose an API, make a request, and put the data into a data frame.

API Request

First, I obtained an API key from the above site by creating an account and registering an application. I define my `api_key` in the code box below. I set `echo` equal to FALSE so that it doesn't show on knitted HTML/PDF.

I chose to use both the NYT Most Popular API and Most Shared API.

The Most Popular API returns the top viewed articles in the amount of time you specify. I chose 30 so that it returns the most viewed articles of the past month. I sent the API request and inserted the response into a data frame by reading the JSON data.

The Most Shared API is similar but instead returns the most shared articles on the medium you specify (I chose Facebook because the other ones seemed to not be working). In my request, I also told it to return the most shared articles by Facebook for the past 30 days. Similar to the one above, I placed that data into a data frame from the JSON data.

```r
library(httr)
library(jsonlite)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter()  masks stats::filter()
## x purrr::flatten() masks jsonlite::flatten()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(ggplot2)
library(knitr)



#Most Viewed API
```

```r
url_view = "https://api.nytimes.com/svc/mostpopular/v2/viewed/30.json"

response_view = GET(url_view, query = list("api-key" = api_key))

parsed_data_view = fromJSON(content(response_view, as = "text"))

#most viewed articles past 30 days
articles_view = parsed_data_view$results



url_shared = "https://api.nytimes.com/svc/mostpopular/v2/shared/30/facebook.json"

response_shared = GET(url_shared, query = list("api-key" = api_key))

parsed_data_shared = fromJSON(content(response_shared, as = "text"))

#most shared fb articles past 30 days
articles_shared = parsed_data_shared$results
```

Data Cleaning

Here I clean the data collected using Tidyverse. I clean the data to attempt to answer some questions I made up that could be answered with the data.

Most Popular Sections

Which Section in the NYT's were most represented by the Most Viewed articles in past month and most shared on Facebook in past month?

```r
#names(articles_view)

#putting in data frame and getting the columns we want

#top 20 articles viewed this past month
articles_view_dat= articles_view %>%
  select(id, title, section, subsection, type, published_date, updated, geo_facet, url)

#top 20 facebook articles viewed this past month
articles_shared_dat= articles_shared %>%
  select( id, title, section, subsection, type, published_date, updated, geo_facet, url)
```

```r
#In the most viewed articles, which sections were the most popular?
most_viewed_sect=articles_view_dat %>%
  arrange(section) %>%
  select(section, everything())

viewed_sect_count=most_viewed_sect %>%
  count(section) %>%
  arrange(desc(n))

top_3_sect= viewed_sect_count %>%
  head(3)
```

```
total_articles_view=sum(viewed_sect_count$n)

cat("The Top three sections represented in the NY Times most viewed articles in the past 30 days are: \n
    top_3_sect$section[1], "with",top_3_sect$n[1], "articles total representing it (", ((top_3_sect$n[1
    top_3_sect$section[2], "with",top_3_sect$n[2], "articles total representing it (", ((top_3_sect$n[2
    top_3_sect$section[3], "with",top_3_sect$n[3], "articles total representing it (", ((top_3_sect$n[3
```

```
## The Top three sections represented in the NY Times most viewed articles in the past 30 days are:
##  U.S. with 9 articles total representing it ( 45 % of all top viewed articles.)
##  Opinion with 5 articles total representing it ( 25 % of all top viewed articles. )
##  World with 3 articles total representing it ( 15 % of all top viewed articles.)
```

```
#Lets compare that to the sections of top facebook articles shared in past 30 days
most_shared_sect=articles_shared_dat %>%
  arrange(section) %>%
  select(section, everything())

shared_sect_count=most_shared_sect %>%
  count(section) %>%
  arrange(desc(n))

top_3_sect_fb= shared_sect_count %>%
  head(3)
total_articles_fb=sum(shared_sect_count$n)

cat("The Top three sections represented in the NY Times most shared Facebook articles in the past 30 da
    top_3_sect_fb$section[1], "with",top_3_sect_fb$n[1], "articles total representing it (", ((top_3_se
    top_3_sect_fb$section[2], "with",top_3_sect_fb$n[2], "articles total representing it (", ((top_3_se
    top_3_sect_fb$section[3], "with",top_3_sect_fb$n[3], "articles total representing it (", ((top_3_se
```

```
## The Top three sections represented in the NY Times most shared Facebook articles in the past 30 days
##  Opinion with 11 articles total representing it ( 55 % of all top viewed articles.)
##  Arts with 2 articles total representing it ( 10 % of all top viewed articles. )
##  Business with 2 articles total representing it ( 10 % of all top viewed articles.)
```

Most Popular Page Type

What page type were the most popular in both the Most Viewed group and the Facebook group?

```
#Which are the most popular article types between the articles shared on fb and most viewed

#In the most viewed articles, which article types were the most popular?

most_viewed_type=articles_view_dat %>%
  arrange(type) %>%
  select(type, everything())

viewed_type_count=most_viewed_type %>%
  count(type) %>%
  arrange(desc(n))

cat("For the Top 20 Viewed NYT pages in the past 30 days, \n",
    viewed_type_count$type[1], "represent the larger group with",viewed_type_count$n[1], "instances or 
    viewed_type_count$type[2], "represent the smaller group with",viewed_type_count$n[2], "instances or
```

```
## For the Top 20 Viewed NYT pages in the past 30 days,
##  Article represent the larger group with 19 instances or consisting of 95 % of Top 20 Viewed
##  and Interactive represent the smaller group with 1 instances or consisting of 5 % of Top 20 Viewed.
```

```
#Lets do the same for the top facebook shared articles
most_shared_type=articles_shared_dat %>%
  arrange(type) %>%
  select(type, everything())

shared_type_count=most_shared_type %>%
  count(type) %>%
  arrange(desc(n))

cat("For the Top 20 NYT pages shared in Facebook in the past 30 days, \n",
    shared_type_count$type[1], "represent the larger group with",shared_type_count$n[1], "instances or
    shared_type_count$type[2], "represent the smaller group with",shared_type_count$n[2], "instances or
```

```
## For the Top 20 NYT pages shared in Facebook in the past 30 days,
##  Article represent the larger group with 20 instances or consisting of 100 % of Top 20 Viewed
##  and NA represent the smaller group with NA instances or consisting of NA % of Top 20 Viewed.
```

Overlap

How many articles overlap between the two groups?

```
#How many articles are the same between the two groups

#going to combine both into 1 df with two new columnsindicating which one they came from

matching_id=inner_join(articles_view_dat,articles_shared_dat, by="id")

cat("There are",nrow(matching_id),"articles in common between the most viewed in past 30 days and most
```

```
## There are 2 articles in common between the most viewed in past 30 days and most sared on Facebook in
##  They are:
```

```
for (i in (1:nrow(matching_id))){
  cat(matching_id$title.x[i], "which was published in the",matching_id$section.x[i], "section on", matc
}
```

```
## Israel Is About to Make a Terrible Mistake which was published in the Opinion section on 2023-10-19
## What They Don’t Tell You About Getting Old which was published in the Opinion section on 2023-09-30
```

Time Difference

Which articles had the greatest amount of time passing from the date it was published to the time it was last updated in both groups?

```
#Out of all articles, which ones have the longest leghth of time between first published and last updat

merged_df= bind_rows(articles_view_dat,articles_shared_dat) %>%
  distinct()

merged_df$published_date=as.Date(merged_df$published_date)
merged_df$published_date=as.POSIXct(merged_df$published_date)

merged_df$updated=as.Date(merged_df$updated)
```

```r
merged_df$updated=as.POSIXct(merged_df$updated)

merged_df= merged_df %>%
  mutate(t_diff=as.numeric(difftime(updated,published_date, unit= "days")))

time_df= merged_df %>%
  arrange(desc(t_diff)) %>%
  select(t_diff, title)

kable(time_df, caption = "Time passed in days between first published and last updated in descending or
```

Table 1: Time passed in days between first published and last updated in descending order

t_diff	title
17	Maps: Tracking the Attacks in Israel and Gaza
12	The Beekeepers Who Don’t Want You to Buy More Bees
6	Scalise Withdraws as Speaker Candidate, Leaving G.O.P. in Chaos
6	Judge Rules Trump Committed Fraud, Stripping Control of Key Properties
5	Known for His Pointed Questions, a 15-Year-Old Is Ejected From a G.O.P. Event
4	The Plot Trump Lost
4	Charles Feeney, Who Made a Fortune and Then Gave It Away, Dies at 92
4	I Study Climate Change. The Data Is Telling Us Something New.
4	We Must Not Kill Gazan Children to Try to Protect Israel’s Children
2	Israel Is About to Make a Terrible Mistake
2	McCarthy Faces Test as Gaetz Moves to Oust Him for Working With Democrats
2	Meet the Republicans Running for Speaker
2	The Secrets Hamas Knew About Israel’s Military
2	What They Don’t Tell You About Getting Old
1	Republican Tempers Flare as Speaker Fight Continues, Paralyzing the House
1	8 Sex Myths That Experts Wish Would Go Away
1	Sidney Powell Pleads Guilty in Georgia Trump Case
1	Gaetz Moves to Oust McCarthy, Threatening His Grip on the Speakership
1	G.O.P. Nominates Mike Johnson for Speaker After Spurning Emmer
1	Why Israel Is Acting This Way
1	How Israel’s Feared Security Services Failed to Stop Hamas’s Attack
1	Israel Has Never Needed to Be Smarter Than in This Moment
1	Man Is Charged With Murder in Tupac Shakur Case
1	Barnes & Noble Sets Itself Free
1	Burt Young, ‘Rocky’ Actor Who Played Complex Tough Guys, Dies at 83
1	Rudolph Isley, an Original and Enduring Isley Brother, Dies at 84
1	President Biden’s Finest Hour
1	I’m Going to War for Israel. Palestinians Are Not My Enemy.
1	Nobel Prize Awarded to Covid Vaccine Pioneers
1	My Fellow Republicans: It’s Time to Grow Up
1	David McCallum, Heartthrob Spy of ‘The Man From U.N.C.L.E.,’ Dies at 90
1	Can We Talk About Joe Biden?
1	Suzanne Somers, Star of ‘Three’s Company,’ Is Dead at 76
1	Jenna Ellis, Former Trump Lawyer, Pleads Guilty in Georgia Election Case
0	The senator was hailed as a pioneer in politics. Here’s what to know.
0	What Does Destroying Gaza Solve?
0	One Reason the Trump Fever Won’t Break
0	Hail to the Fraudster in Chief

```r
cat("The Top three articles from the most viewed and facebook group that had the most time passed from f

    time_df$title[1], ":\n with",time_df$t_diff[1],  "days passing since the article was originally publ

    time_df$title[2], ": \n with",time_df$t_diff[2],  "days passing since the article was originally pu

    time_df$title[3], ":\n with",time_df$t_diff[3], "days passing since the article was originally publi
```

```
## The Top three articles from the most viewed and facebook group that had the most time passed from fi
##
##  Maps: Tracking the Attacks in Israel and Gaza :
##  with 17 days passing since the article was originally published.
##
##  The Beekeepers Who Don't Want You to Buy More Bees :
##  with 12 days passing since the article was originally published.
##
##  Scalise Withdraws as Speaker Candidate, Leaving G.O.P. in Chaos :
##  with 6 days passing since the article was originally published.
##
```

Conclusion and Future Directions

The New York Times has a lot of cool and different APIs people can use to accomplish a variety of tasks. I chose the Most Viewed and Most Shared APIs to simulate building a report on the most viewed/shares articles on the NYT Site. You can possibly use this report to analyze the reasons why some articles are getting more clicks or some topics or more interesting than others. A future direction I would like to head in is to analyze what the most shared articles look like between the different social media platforms (X, Reddit, Facebook). You can do this to see differences in articles that are interested to the population of each media platform. Another interesting thing to do is to run the most viewed report everyday for a period of a month and see how long the articles remained the top viewed articles. APIs are an important interface that helps facilitate the transfer of data so that people can use it in processes. Any changes to APIs by the company (lets say the structure of the Most Viewed API were to change) cause processes that are dependent on this API to break.