Automatic Sampling and Analysis of YouTube Data

The YouTube API

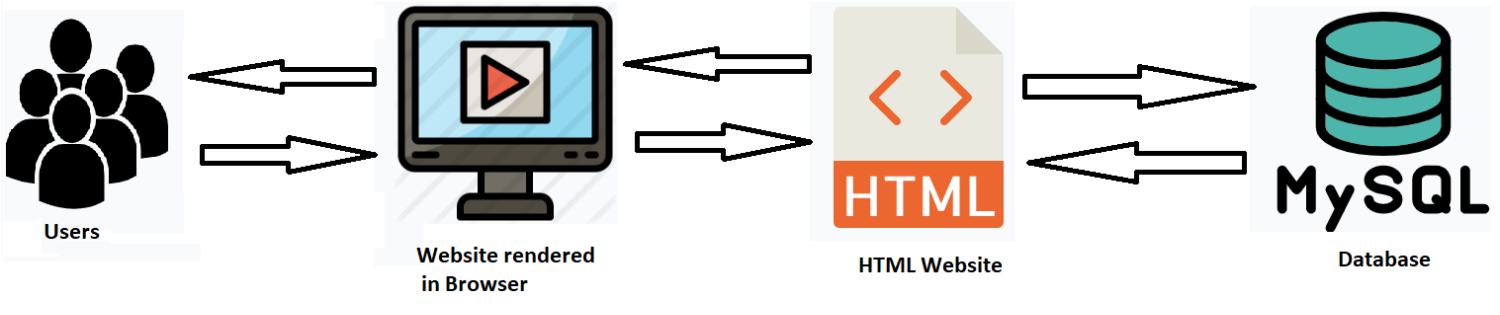
Julian Kohne
Johannes Breuer
M. Rohangis Mohseni

2020-02-10

The YouTube API

Overview

- All data on YouTube is stored in a MySQL database
- The website itself is an HTML page, which loads content from this database
- The HTML is rendered by a webbrowser so the user can interact with it
- Through interacting with the rendered website, we can either retrieve content from the database or send information to the database
- The Youtube Websi!te is
    - built in HTML,
    - uses CSS for the "styling"
    - dynamically loads content using Ajax from the Database



How do we get Data From Websites?

- Theoretically, we could gather all the information manually by clicking on the things that are interesting to us and copy/pasting them. However, this is tedious and time-consuming. **We want a way of automatizing this task**

- Webscraping

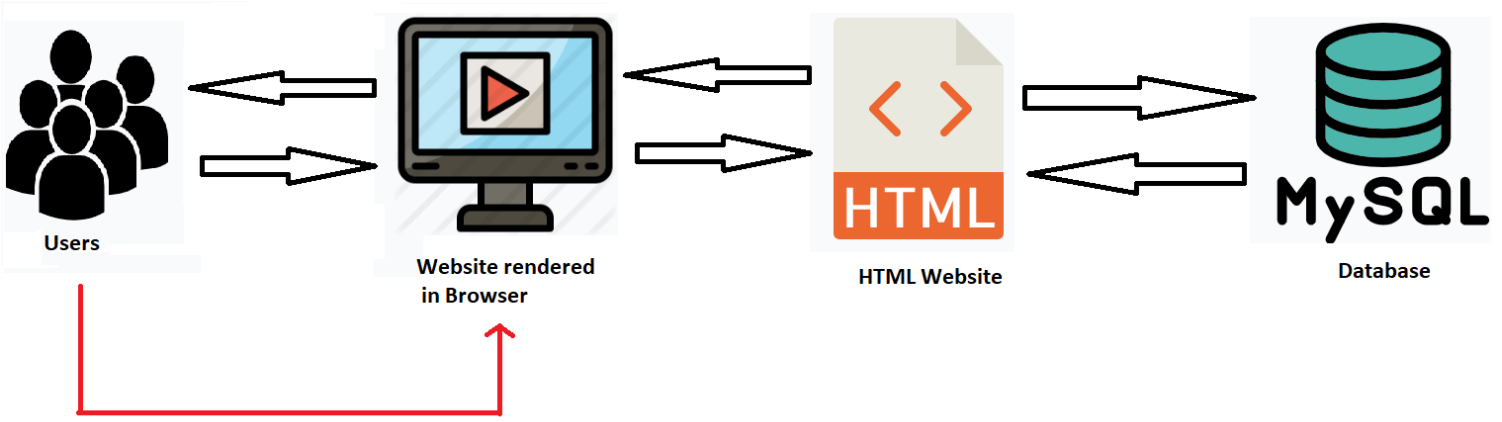
  1) **Screenscraping:** Getting the HTML-code out of your browser, parsing & formatting it, then analyzing the data

  2) **API-harvesting:** Sending requests directly to the database and only getting back the information that you want and need.

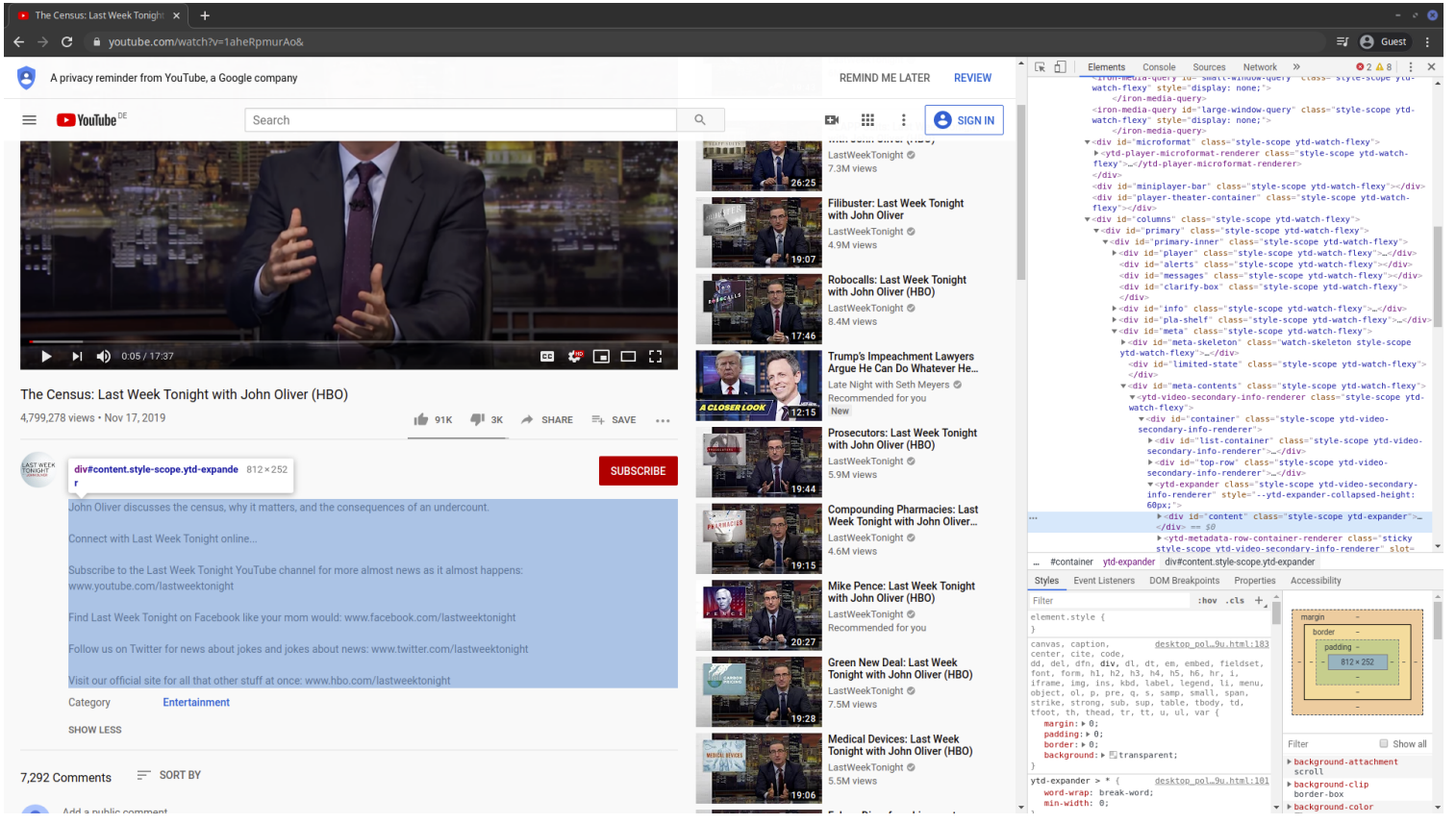
Screenscraping

Screenscraping

- Screenscraping means that we are downloading the HTML textfile, which contains the content we are interested in but also a lot of unnecessary clutter that describes how the website should be rendered by the browser



Screenscraping



Screenscraping

- To automatically obtain data, we can use a so called GET request
- A GET request is an HTTP method for asking a server to send a specific resource (usually an HTML page) back to your local machine
- You can try it out in your console
- This is the basic principle that all the Scraping packages build-on
- We will not use this directly and will let the higher-level applications handle this under the hood

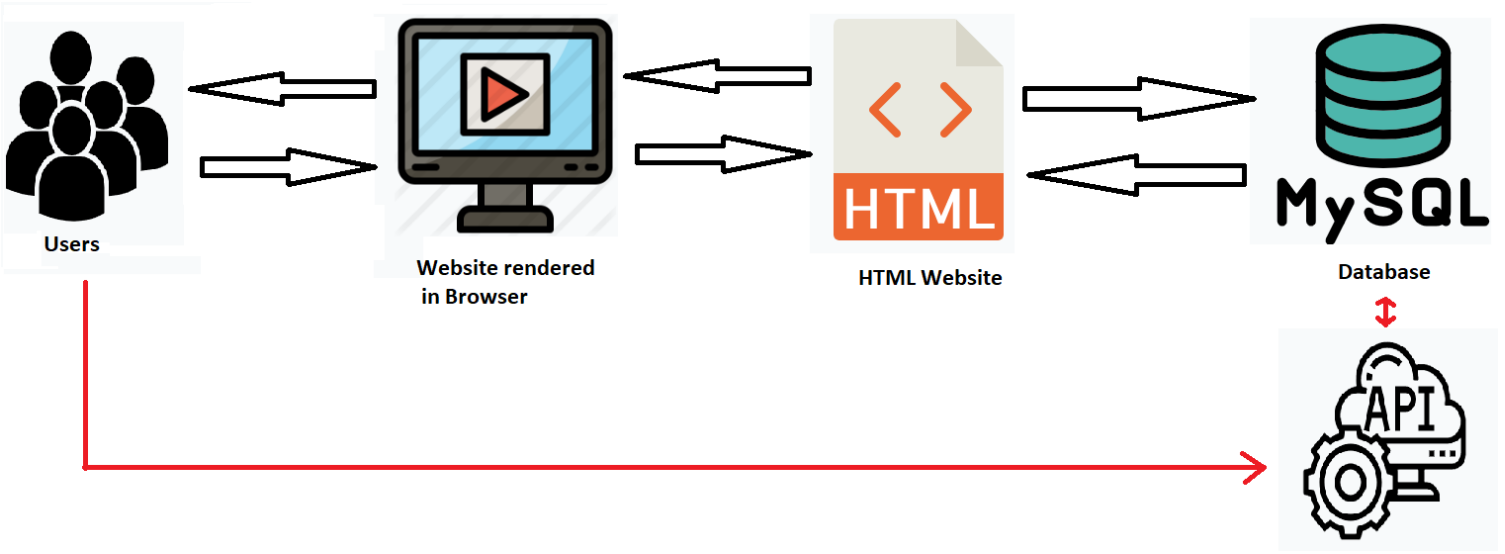
Screenscraping

- Advantages of Screenscraping:
  - You can access everything that you are able to access from your browser
  - You are (theoretically) not restricted in how much data you can get
  - (Theoretically) Independent from API-restrictions
- Disadvantages of Screenscraping:
  - Extremely tedious to get information out of HTML-pages
  - You have to manually look up the Xpaths/CSS/HTML containers to get specific information
  - Reproducibility: The website might be tailored to stuff in your Cache, Cookies, Accounts etc.
  - There is no guarantee that even pages that look the same have the same underlying HTML structure
  - You have to manually check the website and your data to make sure that you're getting what you want
  - If the website changes anything in their styling, your scripts won't work anymore
  - Legal Gray Area (recent court ruling though)

API-Harvesting

API-Harvesting

- An Application Programming Interface) is:
  - a system build for developers
  - directly communicating with the database
  - Voluntary service of the website
  - dictating what information is accessible, to whom, how, and in which quantities.



API-Harvesting

- APIs can be used to:
  - embed content in other applications
  - create Bots that do something automatically
  - scheduling/moderation for content creators
  - collect data for (market) research purposes
- Not every website has their own API. However, most large Social Media Websites do
  - Facebook
  - Twitter
  - Instagram
  - Wikipedia
  - Google Maps

API-Harvesting

- Advantages of API-Harvesting:
    - No need to interact with HTML files, you only get the information you asked for
    - The data you get is already nicely formatted (usually JSON files)
    - You can be sure that what you do is legal and (probably) in line with Terms of Service
- Disadvantages of API-Harvesting:
    - Not every website has an API
    - You can only get what the it allows you to get
    - There are often restricting quotas (e.g. daily limits)
    - there is no standard language to make querys, you have to check the documentation
    - Not every API has a (good) documentation

Screenscraping vs. API-Harvesting

If you can, use an API, if you must, use Screenscraping instead

The YouTube API

Summary

- Fortunately, YouTube has their own, well-documented API that developers can use to interact with their database (Most Google Services do)
- To find an API for a given website, Programmable Web is a good starting point
- We will use the YouTube API today

Let's Check the API

- Google provides a sandbox for their API that we can use to get a grasp of how it operates
- We can for example use our credentials to get search for videos with the keyword "Brexit"
- Example
- Keep in mind: We have to log in with our created Google account to use the API
- What we get back is a JSON formatted response with the formats and information we requested in the Sandbox

What is JSON?

- Java Script Object Notation
- Language indpendent data format (like .csv)
- Like a nested List of Key:Value pairs
- Standard data format for many APIs and web applications
- Better than tabular formats (.csv / .tsv) at storing large quantities of data by not declaring missing data
- Represented in R as a list of lists, needs to be transformed into a regular dataframe (this can be tedious)

What is JSON?

```
'{
  "first name": "John",
  "last name": "Smith",
  "age": 25,
  "address": {
    "street address": "21 2nd Street",
    "city": "New York",
    "postal code": "10021"
  },
  "phone numbers": [
    {
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "mobile",
      "number": "646 555-4567"
    }
  ],
  "sex": "male"
}'
```

Most Important Parameters

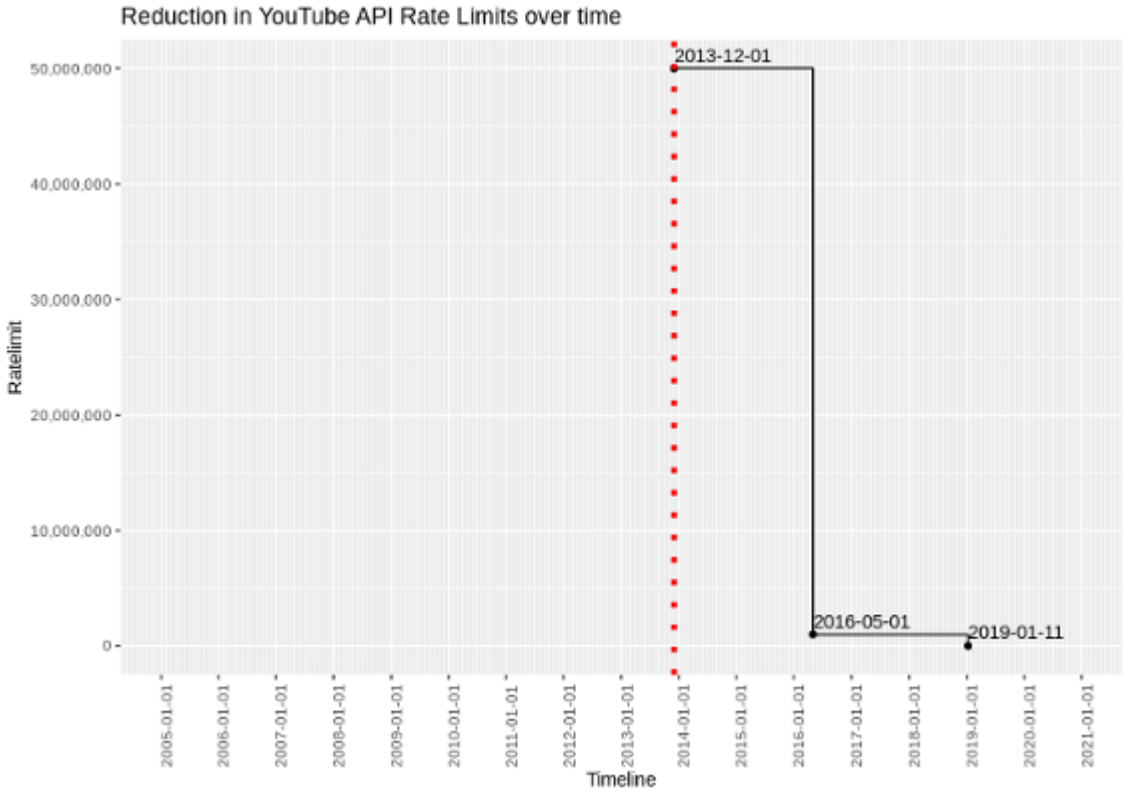
- All possible parameters are listed here
- Keep in mind that some information is only visible to owners of a channel or author of a video
- Keep in mind that not all information is necessarily available for all videos (e.g. live videos)

Using it from R

- We can simplyfiy the process of interacting with the YouTube API by using a dedicated R package
- The package handles the authentification with our credentials and translates R commands into API calls
- It also simplifies the JSON response to a standard dataframe automatically
- In essence, we can run R commands and get nicely formatted API results back
- For this workshop, we will thus use the tubeR packageUsing it from

Rate Limits

- With the API, you have a limit of how much data you can get
- This limit has constantly decreased over the last decade



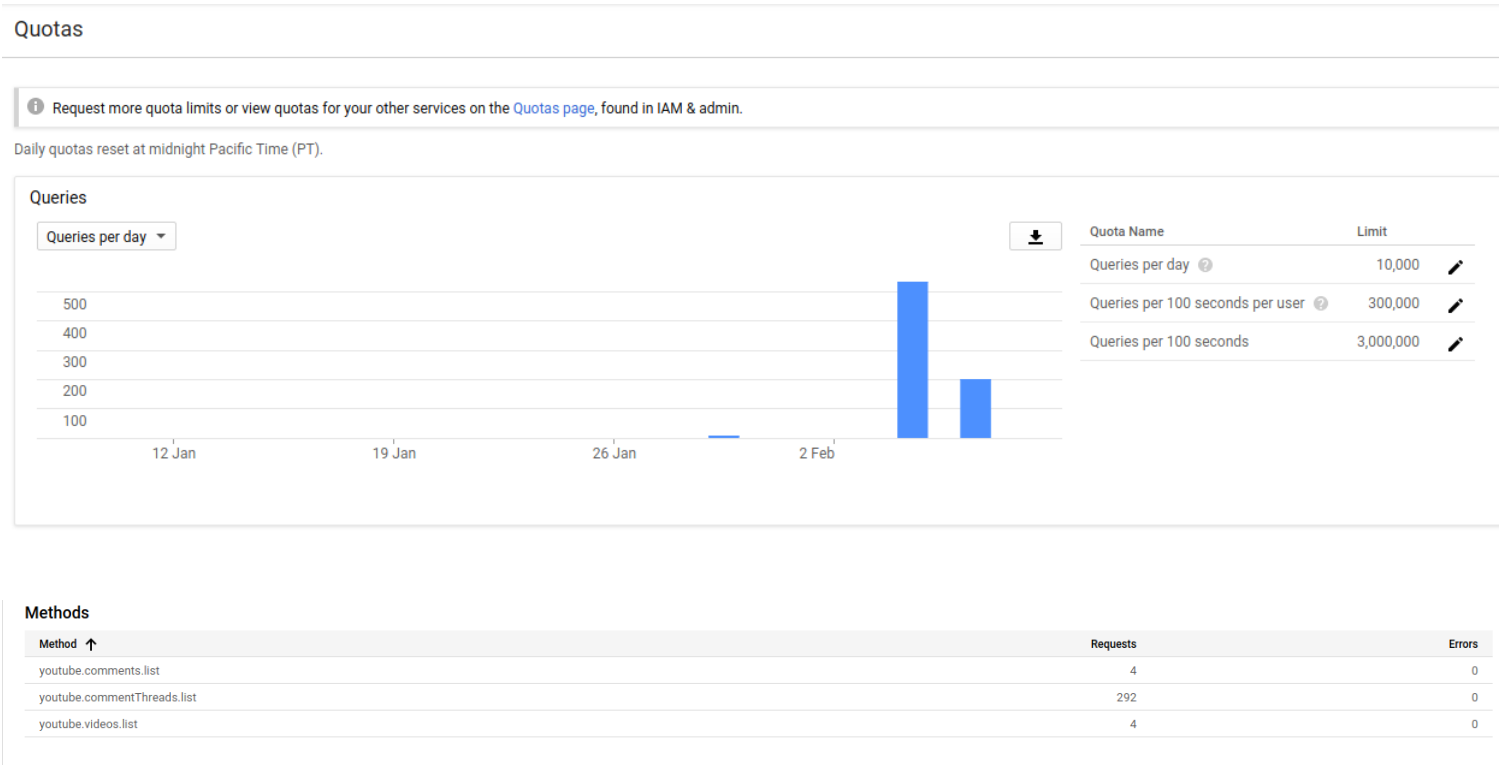
Rate Limits

- Currently (02.2020), you have a quota of **10.000** units per day
- Each request (even invalid ones) costs a certain amount of units
- There are two factors influencing the quota cost of each request:
  - different types (e.g write operation: 50 units; video upload: 1600 units)
  - how many parts the requested resource has (playlist:2 ; channel:6 ; video:10)
- **You should only request parts that you absolutely need to make the most of your units. More on that in the data collection session.**

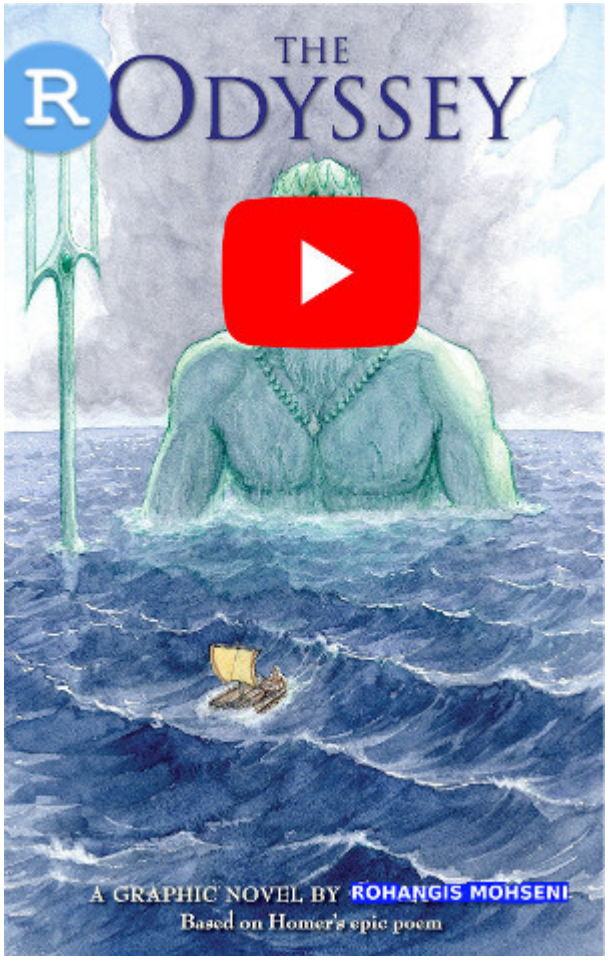
BEWARE: Sending wrong requests can fill up your daily quota

Rate Limits

- You can check the rate limits in the *YouTube* API Documentation
- You can see how much of your quota you have already used up in the developer console



Can I Increase my Rate Limit?



Raising the Quota Limit for YouTube

- Study planned that needs large datasets in short amounts of time
- RQ: Is there a u-shaped relationship between success and number of uploads?
- Sample: 600 popular channels (identified via SocialBlade)
- Request for higher quota (October 11, 2019)
- Problem: Same application form for (web) apps and research
- Hard to figure what applies to research and what to write into the form

31/10/2019 First Response

- Provide us with the information from where the 600 accounts have been acquired?
- How many accounts from YouTube are used for your research?
- Kindly provide us a visual reference of how the data is sampled and analysis is performed on the respective accounts.
- Answer in 3 business days.

31/10/2019 My (Shortened) Reply

- We want to investigate if the success of YouTube “channels” changes over time. For this purpose, we will create a list of 600 channels (“social media stars”) in order to compare them with each other. We will probably use a ranking service like Socialblade for that purpose.
- We are using RStudio for the data collection. Within RStudio, we installed an R package called “tuber”. Tuber offers certain functions to collect data via the YouTube API. Tuber can connect via OAUTH if it is provided with credentials. We want to use the credentials from my personal Google account for that purpose. Thus, a single account will be used.
- I am not sure what you mean by “visual” reference. We want to use the tuber package within R for data sampling by calling the following tuber functions for each of the channels that shall be investigated: get_channel_stats, get_video_details, and get_all_channel_video_stats. We also want to use R for data analysis. We had another project in which we created an R script that can analyze YouTube comments. It is “visualized” in form of a Jupiter notebook. Maybe this can illustrate what we want to do for our current project. However, the new code will be way less complex, but will be run once a week in order to collect a time series.

11/11/2019 Second Response

- How is the Data being compared and how is the Ranking service performed?
- What kind of Analytics are being performed on the API client website?
- Please provide us with detailed screenshots and screencast of the API use case.
- Answer in 3 business days.

11/11/2019 My Reply

- The comparisons that we want to perform are statistical tests (descriptive and inferential statistics) that we conduct in order to answer our research questions. For instance, we want to know if the popularity of the channels depends on the number of uploads. We would gather data on the number of video uploads, the number of video likes, channel likes etc. with the tuber package, and then perform statistical tests like multiple linear regressions. You can do this with any statistical program like SPSS, Stata etc. We will use R for that purpose because R can also collect the data we need via the tuber package.

- In order to figure from which channels we want to collect data, we will probably use some pre-existing ranking service like SocialBlade. We will go to their website (socialblade.com), check out some of the TOP channel lists (https://socialblade.com/youtube/top/500), and manually create a spreadsheet of 600 channels that may be of interest to us. We will then fill in the YouTube IDs of the channels. The IDs will help us to collect the data via the tuber package.

- There is no such thing as a website. The API client “tuber” is a package within the (Windows/Linux/Mac) desktop program “RStudio”. The data we collect is never displayed via any service to anyone. It will only be used to perform the statistical tests.

12/11/2019 Third Response

- Violation: API Client is storing data for more than 30 days.
- Fix in 3 business days.

12/11/2019 My Reply

- I am not sure how to comply with the policy
    - because I cannot tell for sure what you consider to be the API client and
    - because we did not implement anything by now (we are talking about a future implementation).
- Right now, our planned pipeline looks like this YouTube data -> YouTube data API -> tuber package (a “plugin” for R that runs in R) -> RStudio (a desktop program for statistical analyses).
- Would you consider tuber or RStudio to be the API client? If you think RStudio is the API client, would it suffice to automatically remove the data from RStudio within 30 days?

18/12/2019 Fourth Response

- Please share a detailed step by step screencast (Video recording) of how the analysis of channels are performed along with the end result to verify the use case.
- Answer in 3 business days.

My Reply

- Guys, it's holiday season (just a thought, no actual reply)

06/01/2020 Fifth Response

- Please share a detailed step by step screencast (Video recording) of how the analysis of channels are performed along with the end result to verify the use case.
- Answer in 3 business days.

06/01/2020 My Reply

- You can download the requested screencast using the following link:

13/01/2020 Sixth Response

- Violation: API Client is storing data for more than 30 days.
- Fix in 3 business days.

13/01/2020 My Reply

- The exact same issue was already raised by you in November. I already replied to it, but I never got a response. Therefore, could you please be more specific to what you actually mean with “API client”?
- As stated in the screencast, our planned pipeline looks like this:
    - YouTube data -> YouTube data API -> tuber package (a “plugin” for R that runs in R) -> RStudio (a desktop program for statistical analyses which could be considered the API client) -> transfer the data from the computer that downloaded it to a computer that will run the statistical tests (in another RStudio). This will be done once per week if possible so the data stored on the “API client” will be overwritten anyway.
    - Therefore, the data would not remain very long on the computer that downloaded it, probably not more than one day. Also, it is not accessible by anyone but us anyway.
- If you do not think this is sufficient, could you please explain in more detail why this is the case?

13/01/2020 Sixth Response

- Violation: API Clients must not replace API Data with similar, independently calculated data, or use API Data to create new or derived data or metrics
    - Providing ranking to channels based on views,subscribers,comments on a sample data from 600 accounts on a weekly base to figure if the number of weekly uploads impacts the popularity of the account.
- Fix in 3 business days.

21/01/2020 My Reply (1)

- Please excuse the delayed reply. I had to think a little bit about how to reply because the correct short answer “we don’t replace data” will probably be not very helpful to you.
- Therefore, I will again try to explain what we want to do and why this request (and probably most of the other compliance rules) do not make sense in our use case.

1. We want to conduct a scientific study. That means that we do not want to provide any service of any kind.
2. We are using the website Socialblade to see which channels are popular because we want to investigate popular channels.
3. We do not alter Socialblade rankings or create our own rankings based on Socialblade. We only need it to identify which channels are currently popular.
4. For our study, we now want to collect data about these channels, like the number of video uploads, channel likes, video likes and so own.
5. With this data, we want to investigate if the number of uploads has some impact on the success of the channel. This is an important academic question.

21/01/2020 My Reply (2)

6. We need to collect the data via the API because collecting it manually will be practically infeasible due to the large sample size of 600 channels.
7. We could collect the data with different tools, for instance, YouTube Data Tools (https://tools.digitalmethods.net/netvizz/youtube/), Facepager (https://github.com/strohne/Facepager), or the tuber package in R.
8. We decided to use the tuber package because we want to sample the 600 channels on the same day and ideally without any larger time differences between each channel.
9. Therefore, it is again very unpractical to use anything but the tuber package because YouTube Data Tools and Facepager would give us more than 600 files (at least one for each channel) each time, and the sampling with these tools would introduce time differences because it cannot be completely automated .
10. If we would use YouTube Data Tools, you would not demand any form of compliance from us.
11. The interesting thing is now that R cannot only collect the data, but also do all the necessary significance tests.

21/01/2020 My Reply (3)

12. Significance tests do not alter the data, they test if a certain pattern in the data exists that cannot be explained by chance alone (https://en.wikipedia.org/wiki/Statistical_significance), for instance, a u-shaped relationship between two variables.
13. So again, if we would use YouTube Data Tools to collect the data and then perform the significance tests, no form of compliance would be requested from us.
14. Coming back to your request, you ask that “API Clients must not replace API Data with similar, independently calculated data, or use API Data to create new or derived data or metrics”.
15. If you now believe that significance tests are violating this request, you actually prohibit ANY KIND OF RESEARCH based on data that was collected with the API.
16. This does not seem to make sense because (a) providing a form for research requests would not make sense, (b) this rule is not enforced when using other tools like YouTube Data Tools or Facepager, (c) significance tests do not alter the data, and (d) significance tests do not provide the data to anybody like a service would.

21/01/2020 My Reply (4)

17. Also, one could argue that after sampling the data, we are no longer within the scope of the API client (that is why I asked two times what you define as the API client in our use case) because if we would sample the data with YouTube Data Tools and then analyze it in R, YouTube Data Tools would be the API client, but R would not. We could actually use a completely different statistical software package like SPSS or STATA and perform the same significance tests.
18. I assume that your request does only make sense if we are talking about an API client that is providing a service (like Socialblade). But we do not want to provide such a service. We only want to analyze the data for ourselves.
19. This is why we filled in the form for research requests. However, I have the feeling that our research request is constantly confused with some kind of web service request.

Any questions?

Exercise time □♀□□□

Solutions