Social Media Bonanza

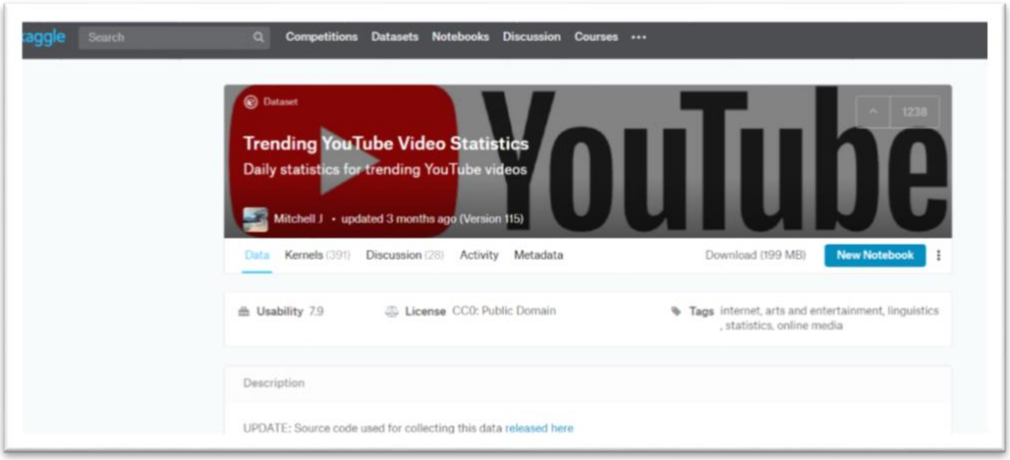
FINAL REPORT

Blake Freeman, Jill Smith, Tomeka Morrison, Trong Nguyen
| ELT Project | 8/25/2019

Extract

After brainstorming several ideas, we as a group decided that we wanted to see the relationship between news and social media. Basically, if a particular trending category in the news has any effect on a trending topic on YouTube.

For this project, our main YouTube data was extracted from *Trending YouTube Video Statistics* on the Kaggle website. The raw data came in the form of a .csv file that was downloaded and then pulled into Pandas. We also found the most popular YouTube API video category from a user called “dgp” on GitHub Gist.



Trending YouTube Video Statistics

Daily statistics for trending YouTube videos

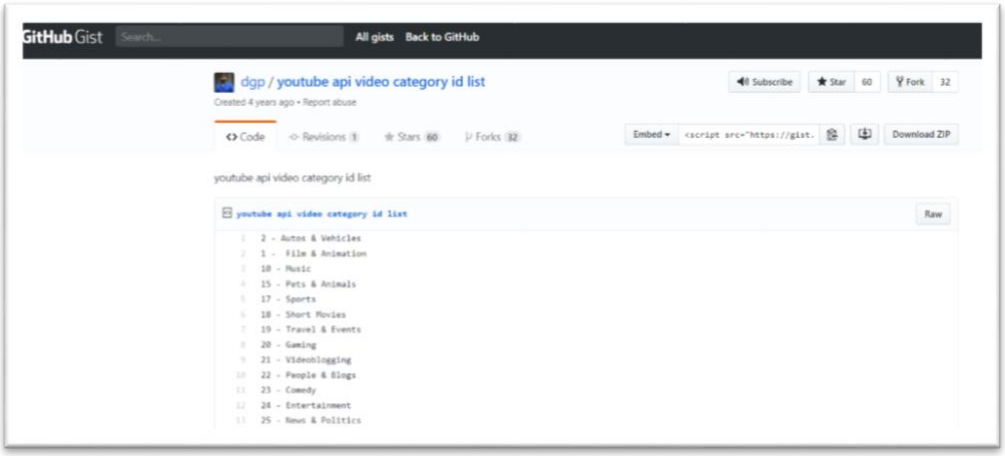
Mitchell J • updated 3 months ago (Version 115)

Usability 7.9

License CC0: Public Domain

Tags internet, arts and entertainment, linguistics , statistics, online media

UPDATE: Source code used for collecting this data released here



dgp / youtube api video category id list

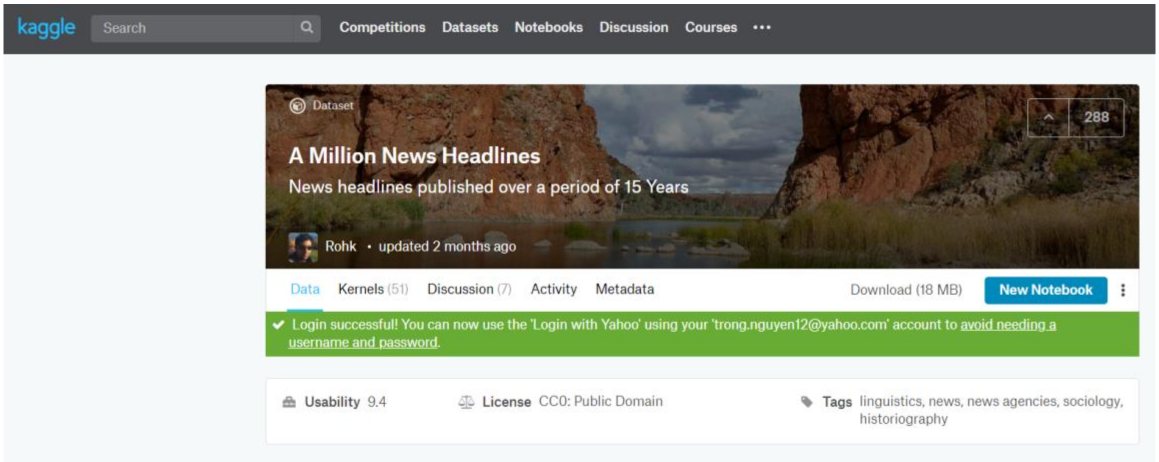
Created 4 years ago • Report abuse

youtube api video category id list

youtube api video category id list

```
2 - Autos & Vehicles
1 -  File & Animation
10 - Music
15 - Pets & Animals
17 - Sports
18 - Short Movies
19 - Travel & Events
20 - Gaming
21 - Videoblogging
22 - People & Blogs
23 - Comedy
24 - Entertainment
25 - News & Politics
```

For our trending news headline, our second data source was pulled from *A Million News Headlines* on the Kaggle website. The raw data came in the form of a .csv file that was downloaded and then pulled into Pandas.



A Million News Headlines

News headlines published over a period of 15 Years

Rohk • updated 2 months ago

✓ Login successful! You can now use the 'Login with Yahoo' using your 'trong.nguyen12@yahoo.com' account to avoid needing a username and password.

Usability 9.4

License CC0: Public Domain

Tags linguistics, news, news agencies, sociology, historiography

- Need a heading? On the Home tab, in the Styles gallery, just click the heading style you want.
- Notice other styles in that gallery as well, such as for a quote, a numbered list, or a bulleted list like this one.
- For best results when selecting text to copy or edit, don't include space to the left or right of the characters in your selection.

TRANSFORM

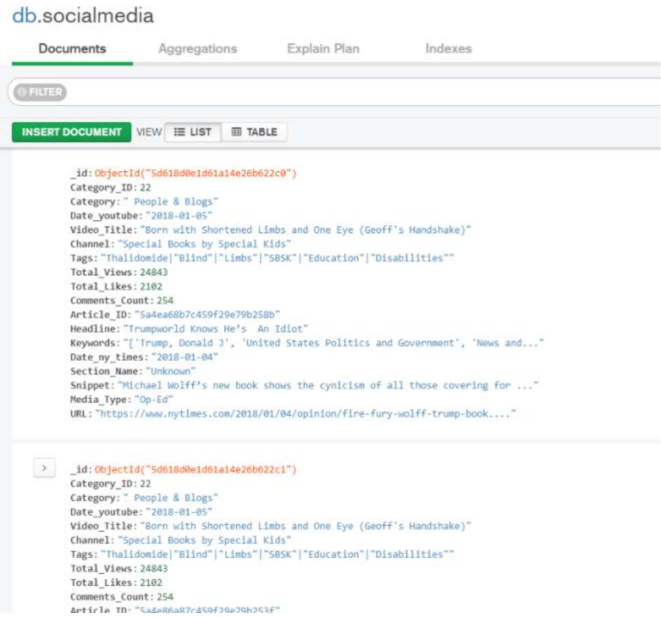
We knew that in order for our analysis work, we had to first transform the raw data from the .csv into a dataset that made sense. Using the Jupyter Notebook, we cleaned the three datasets by converting the time and date to just dates. And then from there, we filteored out the dates that we wanted to analyze. We found that from 1/4/2018 – 05/01/2018, the dates in both the YouTube data and the News data both overlapped each other.

Next, there were a few categories in the News dataset that didn't match up with the YouTube dataset, so we recategorizes them as best as we could. For example, the category metro or metropolitan about local news became just News & Politics.

The cleaned YouTube dataset and the cleaned News dataset was then combined and further transformed by dropping any columns that we thought were not needed. We also formatted some of the column headers to make the whole data frame look better.

LOAD

To bring everything home, we loaded our final database into MongoDB. In the Jupyter Notebook, we imported pymongo and made a connection to the MongoClient. This took us a few tries, but we finally got it to work.



```
_id: ObjectId("5d618d0e1d61a14e26b622c0")
Category_ID: 22
Category: " People & Blogs"
Date_youtube: "2018-01-05"
Video_Title: "Born with Shortened Limbs and One Eye (Geoff's Handshake)"
Channel: "Special Books by Special Kids"
Tags: "Thalidomide|"Blind"|"Limbs"|"SBSK"|"Education"|"Disabilities""
Total_Views: 24843
Total_Likes: 2102
Comments_Count: 254
Article_ID: "5a4ea68b7c459f29e79b258b"
Headline: "Trumpworld Knows He’s  An Idiot"
Keywords: "['Trump, Donald J', 'United States Politics and Government', 'News and..."
Date_ny_times: "2018-01-04"
Section_Name: "Unknown"
Snippet: "Michael Wolff’s new book shows the cynicism of all those covering for ..."
Media_Type: "Op-Ed"
URL: "https://www.nytimes.com/2018/01/04/opinion/fire-fury-wolff-trump-book...."

> _id: ObjectId("5d618d0e1d61a14e26b622c1")
Category_ID: 22
Category: " People & Blogs"
Date_youtube: "2018-01-05"
Video_Title: "Born with Shortened Limbs and One Eye (Geoff's Handshake)"
Channel: "Special Books by Special Kids"
Tags: "Thalidomide|"Blind"|"Limbs"|"SBSK"|"Education"|"Disabilities""
Total_Views: 24843
Total_Likes: 2102
Comments_Count: 254
Article_ID: "5a4e86a87c459f29e79b253f"
```

TECHNICAL ANALYSIS BREAKDOWN

YouTube:

https://www.kaggle.com/datasnaek/youtube-new - Main Data

https://gist.github.com/dgp/1b24bf2961521bd75d6c - category ID's

News Articles:

https://www.kaggle.com/therohk/million-headlines/downloads/million-headlines.zip/8 - Main Data

Clean_News_Articles (Jupyter notebook)

- Import articles csv files using pandas and create dataframes.

    - Files used: ArticlesJan2018, ArticlesFeb2018, ArticlesMarch2018, ArticlesApril2018
- Delete extraneous columns (refer to line 6)
- Append all four dataframes togehter to create a single dataframe
- Explore updated dataframe with function .count
- Groupby New_Category to pull out all the category names and rename categories. (refer to lines 13 to 22)
- Saved cleaned dataframe as csv file

YouTube_Clean (Jupyter notebook)

- Import YouTube csv and Category_ID csv
- Use pandas to create dataframes from the csv files
- Use pandas to merge the dataframes on the category_id
- Pull required columns from dataframe (refer to line 8)
- Use Datetime to update the 'trending_date' column into datetime format
- Update the column names (refer to line 9)
- Save cleaned dataframe as csv file

Social_Media_Bonaza (Jupyter Notebook)

- Import clean YouTube_Clean and Clean_News_Artilces_v2 csv files
- Use pandas create dataframes from csv files
- Use pandas merged the dataframes on 'Category"
- Use JSON to convert the merged dataframe to a dictionary
- Use pymongo added the merged dataframe to MongoDB