

YouTube Viewership as Influenced by Social Media Presence

Trevor Harness - V00867541

Akash Charitar - V00875728

UVic SENG 474 Fall 2019

Introduction

YouTube as a video sharing platform allows monetization by content creators in the form of advertisements presented to the consumer when watching a video. The more views a creator receives on their videos, the more ads are presented and money the creator is awarded. Even for videos which are not monetized, or creators who do not wish to feature any ads, views are important. Views can provide brand exposure, promote works outside of YouTube, or simply validate a hobby. Therefore, a content creator needs to find ways to generate more views on their videos.

An obvious way to generate recognition and attention in the modern world is through social media. Some of the biggest platforms popular among Youtubers today are Instagram and Twitter. These services provide fast communication to a large audience and form personality of the content creator that interacts with their followers. We are interested in how the number of posts, followers, and likes on these platforms may influence the total viewership of Youtuber's channels.

Another interesting phenomenon is often noticed when examining viral videos. These are videos that receive a disproportionate amount of views relative to the uploader's channel size or subscriber count. A small channel may have only a single video with a notable number of views. In some cases, an unpopular content creator could upload a video that receives millions of views when they have almost no subscribers. This is a pattern that we would also like to explore.

We intend to use regression as well as classification algorithms to try to understand which aspects of social media are most important for total video viewership. However, as the majority of this project was spent on data collection and compilation, our main goal is to evaluate the usefulness of our dataset for future analysis.

Dataset

No existing dataset contains all the data we wished to use for our analysis. We provide here an example of a YouTuber and the information we needed to compile. "TheEllenShow" has 10,990 uploads, 35.3 millions subscribers and 17,939,915,162 views on YouTube. TheEllenShow also has an Instagram account with 17,939,915,162 followers, 383 following and 8172 posts with includes pictures and videos. On Twitter, TheEllenShow has an account named "Ellen DeGeneres" with 79050757 followers, 20621 tweets and 1880 likes.

This information is represented in the Dataset.csv file as followed:

TheEllenShow;10,990;35200000.0;17,939,915,162;79800000.0;383;8,172;79050757;20621;1880









1. YouTube Username
2. Number of uploads on YouTube
3. Number of subscribers on YouTube
4. Total number of views of YouTube

5. Number of followers on Instagram
6. Number of following by user on Instagram
7. Number of posts of Instagram
8. Number of followers on Twitter
9. Number of tweets on Twitter
10. Number of likes on Twitter

Data Collection

Manually selecting YouTube accounts and looking their perspective Instagram and Twitter accounts would be too tedious and time consuming. Ideally, we want to analyze the most successful YouTubers, that are channels with the highest number of subscribers. <https://socialblade.com/> is a real-time database of social media platforms that contains the top 5000 most subscribed YouTube channels. We used this as the base for our dataset.

Pulling data from Socialblade

7th	A++		Cocomelon - Nursery Rhymes	478	66.1M	44,366,818,458
8th	A		5-Minute Crafts	3,873	62.2M	16,395,185,659
9th	A++		SET India	33,875	59.9M	43,786,619,758
10th	A+		Canal KondZilla	1,246	53.4M	27,346,791,451
11th	A+		WWE	44,781	51.4M	37,502,571,344
12th	A		Dude Perfect	219	47.5M	9,388,157,458
13th	B		Justin Bieber	135	47.4M	638,057,332
14th	A+		Zee Music Company	4,412	46.6M	21,691,279,224
15th	A		Badabun	4,936	43.4M	14,971,437,574
16th	A		Ed Sheeran	158	42.8M	18,615,307,508

Using the python library BeautifulSoup, we were able to extract the following information: the channel's name, number of uploads, subscribers count and view counts as shown in the picture above.

This was done by extracting the appropriate 'div' tags and 'attrs' styles from the HTML response the python Request received from Socialblade.

In the next step for each of these accounts, we want to find their corresponding Instagram and Twitter if any. One challenge is finding the ID for each channel. For example, from the above picture, "Cocomelon - Nursery Rhymes" is a channel's name and is not an id/primary key. In order to locate the "about page"

which contains links for other social media accounts, we need the id. For “Cocomelon - Nursery Rhymes”, its channelId is UCbCmjCuTUZos6Inko4u57UQ. The URL for the about page would be: <https://www.youtube.com/channel/UCbCmjCuTUZos6Inko4u57UQ/about>

Finding the channelId

We can find the channelId using the YouTube data API [1].

```
def get_channel_username_API(name):
    os.environ["OAUTHLIB_INSECURE_TRANSPORT"] = "1"
    api_service_name = "youtube" api_version = "v3"
    client_secrets_file = "secret.json" # Get credentials and create an API client flow =
    google_auth_oauthlib.flow.InstalledAppFlow.from_client_secrets_file(
        client_secrets_file, scopes)
    credentials = flow.run_console()
    youtube = googleapiclient.discovery.build(
        api_service_name, api_version, credentials=credentials)
    request = youtube.search().list(
        part="snippet",
        q=name,
        type="channel",
        maxResults=1,
    )
    response = request.execute()
    print(response)
```

This returns a json object and channelId can be found in items[‘channelId’].

In the about page, we can use BeautifulSoup to grab the page content and scrape for all links with ‘a’ tags and href=True to look for the keywords ‘instagram.com%2F’ and ‘twitter.com%2F’. The username will be the word that follows. For example:

```
<a class="yt-simple-endpoint style-scope ytd-channel-about-metadata-renderer"
href="/redirect?event=channel_description&q=http%3A%2F%2Finstagram.com%2Ftheellenshow&SRe9vx9uZBh8MTU3NDk3MTA0MEAxNTc0ODg0NjQw">
```

The Instagram username is theellenshow.

Using these steps, we can build a *username_dataset* with all channels that have both an Instagram and Twitter account. *username_dataset* contains YouTube username, Instagram username and Twitter username as records. Note: It is not possible to get usernames/links for other social media accounts directly via the YouTube data API. These links can only be retrieved from the about page.

Getting Information from Instagram

Extracting the number of posts, followers, and following from Instagram proved to be quite challenging. Instagram prevents users from constantly sending requests by limiting replies and preventing web scraping.

First Attempt:

By adding the `/?__a=1` keyword after the URL, Instagram would return a json file which contains the information we need.

For example: `https://www.instagram.com/theellenshow/?__a=1` would return the relevant information as shown below:

```
{"logging_page_id":"profilePage_18918467","... else,"graphql":{"user":... "edge_followed_by":
{"count":80035565}, "followed_by_viewer": false," edge_follow":{"count":384}, ...
edge_owner_to_timeline_media":{"count":8189,"page_info" ... }
```

where:

```
number of followers would be: json_reply['graphql']['user']['edge_followed_by']['count']
number of following would be: json_reply['graphql']['user']['edge_follow']['count']
number of posts would be: json_reply['graphql']['user']['edge_owner_to_timeline_media']['count']
```

This was simplified by removing unnecessary data.

While this technique worked, we could only retrieve around 150 results before getting blocked resulting in a none return from the python request function.

As mentioned above, it is also impossible to scrape the website using BeautifulSoup without being blocked.

Second Attempt

We referred to "<https://www.benlcollins.com/spreadsheets/import-social-media-statistics/>" that uses a google sheet to extract information from the XML. The following formula :

```
=IMPORTXML(<Cell_with_URL>,"//meta[@name='description']/@content")
```

with URL

`https://www.instagram.com/theellenshow/` would output:

"79.8m Followers, 383 Following, 8,172 Posts - See Instagram photos and videos from Ellen DeGeneres(@theellenshow)". From this, we could extract the number of followers, following and posts and build a dataset for Instagram. It took over 4 days to build this dataset since we were constantly getting blocked and had to swap in between the two methods each time.

Getting Information from Twitter

Twitter allows for web scraping using BeautifulSoup. By making a python get request for the URL: `'https://twitter.com/<username>'` we were able to access the html content and extract information as shown in the next page. We repeated this process for each twitter account and created the Twitter dataset. Generating this dataset took around 3 hours for approximately 2000 usernames.

The final step was merging all the datasets to create the Dataset.csv file using `username_dataset` to link them. This step is straight forward.

For each record in Instagram_dataset
 Youtube_username = Find from username_dataset using Instagram_username
 Twitter_username = Find from username_dataset using Instagram_username
 Youtube_Record = Find from youtube_dataset using Youtube_username
 Twitter_record = Find from twitter_dataset using Twitter_username
 Merge Instagram_record, Youtube_Record and Twitter_Record

The final dataset has around 1200 records.

Note: We also tried including Facebook in our model, but it is impossible to scrape Facebook pages and we were unable to secure a development access key from Facebook.

```
def get_twitter_info(username):
    temp = requests.get('https://twitter.com/'+username)
    bs = BeautifulSoup(temp.text,'xml')
    try:
        follow_box = bs.find('li',{'class':'ProfileNav-item ProfileNav-item--followers'})
        tweets_box = bs.find('li',{'class':'ProfileNav-item ProfileNav-item--tweets is-active'})
        likes_box = bs.find('li',{'class':'ProfileNav-item ProfileNav-item--favorites'})
        num_followers = follow_box.find('a').find('span',{'class':'ProfileNav-value'}).get('data-count')
        num_tweets = tweets_box.find('a').find('span',{'class':'ProfileNav-value'}).get('data-count')
        num_likes = likes_box.find('a').find('span',{'class':'ProfileNav-value'}).get('data-count')
    return num_followers,num_tweets,num_likes
    except Exceptions as e:
        print(str(e))
```

References

[1] Search YouTube Data API Google Developers. (n.d.). Retrieved November 27, 2019, from <https://developers.google.com/youtube/v3/docs/search>.

Data Overview

To get a general picture of the distribution of values in the dataset, several histograms are presented in Figures 1 through 4.

Figure 1: Subscribers

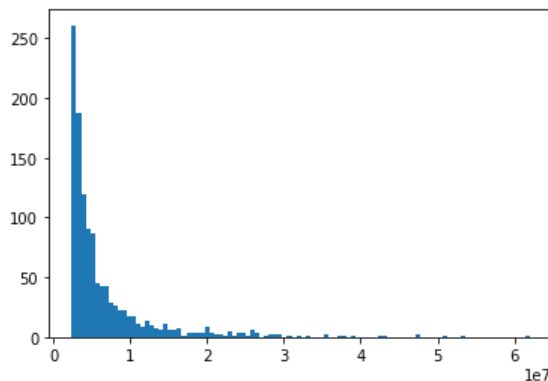


Figure 2: Total Views

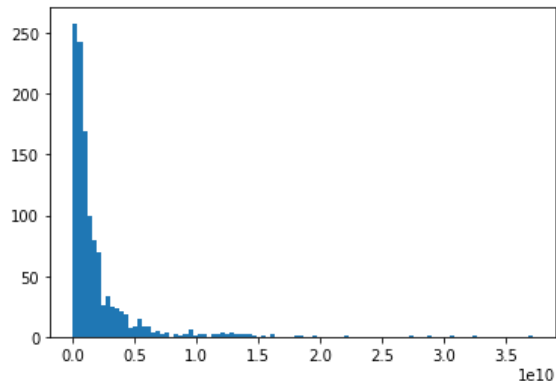


Figure 3: Tweets

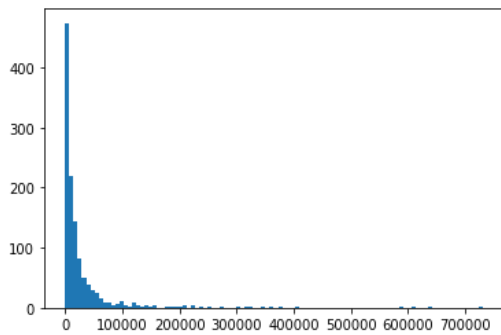
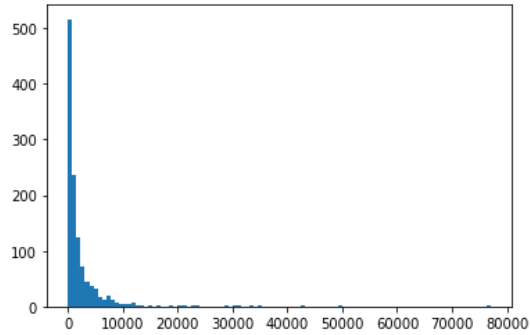


Figure 4: Instagram Posts

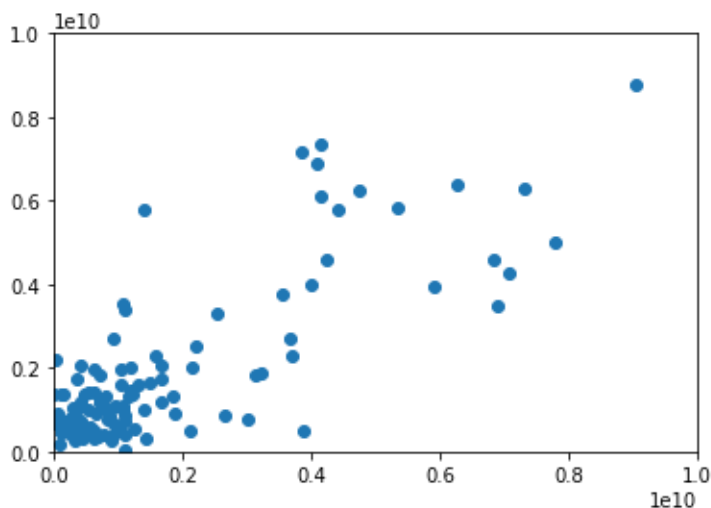


We can see that only very few Youtubers have the largest supporter base, and relatedly, little social media presence. To allow us to run classification on the dataset, we binned total viewership into four categories. These categories correspond to smaller, medium, large, and the largest YouTube channels in terms of total viewership.

Linear Regression

Here we attempt to predict total viewership for a Youtuber based on their social media presence. Experimentation for the most promising features to include led to using number of uploads, subscribers, Instagram posts, Instagram followers, Tweets, and Twitter followers as our features. We ran a Linear Regression model using a data split of 90% training and 10% testing. The target-prediction scatter plot is shown in Figure 5:

Figure 5: Target-Prediction



We see that while the variance in the results is high leading to low accuracy, the model can successfully predict relative viewership. This provides evidence that the weighting of the selected attributes is valid. In order of importance, the attributes and their weights are given in Table 1.

Table 1: Linear Regression Weights

Instagram Posts	Video Uploads	Subscribers	Instagram Followers	Twitter Followers	Tweets
1.06e+05	4.77e+04	3.76e+02	-3.54e+01	-7.65e+01	-4.98e+03

Without looking at the results, it is expected that the number of videos a user uploads has a clear and large impact on the total number of views on their channel. Our model demonstrates this relationship. Compared to this the number of subscribers has a smaller impact. One possible explanation is that not all subscribers watch new videos. According to our model, the number of posts a user makes to Instagram has the largest impact on total video views. This demonstrates the importance of social media presence, and, on a visual platform that itself allows short videos to be published.

Classification

Here we ran classification algorithms to test whether our data can be successfully used to determine the class (small, medium, large, largest) of YouTuber total viewership. If so, our results will give us confidence that there is a predictive relationship between social media presence and success for content creators as measured by views.

We used several models on our binned data and attempted to demonstrate that our dataset contains important and predictive information.

Perceptron

A simple Perceptron classifier was able to achieve 96% accuracy on our test set. However, this result is deceptive. The precision and recall scores for the model are given in Table 2.

Table 2: Perceptron Scores

	Small	Medium	Large	Largest
Precision	0.96035242	0.0	0.0	0.0
Recall	0.95196507	0.0	0.0	0.0

We see that the only values given are for small YouTubers. This problem will be addressed in the classification summary.

Logistic Regression

We were again able to achieve 96% accuracy on our test set. This is unsurprising given the result from a perceptron model.

Table 3: Logistic Regression Scores

	Small	Medium	Large	Largest
Precision	0.97435897	0.0	0.0	0.0
Recall	0.98275862	0.0	0.0	0.0

Decision Tree

A decision tree model achieved 95% accuracy on our test set.

Table 4: Decision Tree Scores

	Small	Medium	Large	Largest
Precision	0.97413793	0.0	0.0	0.0
Recall	0.97413793	0.0	0.0	0.0

Classification Summary

Although the metrics appear promising, they are misled by the nature of the dataset. Most YouTubers are categorized as small channels. The percentage breakdown is seen in Table 5.

Table 5: Class Breakdown

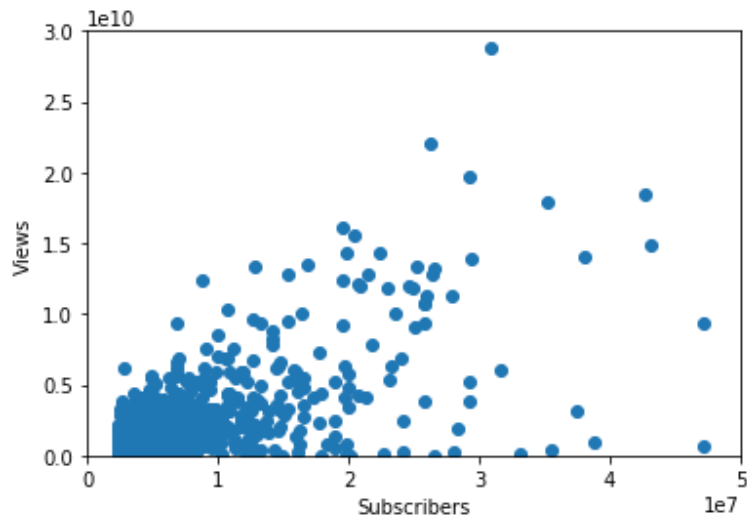
Small	96.3%
Medium	3.1%
Large	0.3%
Largest	0.3%

This disparity between class frequencies makes this type of classification very difficult. These results are unreliable and provided here as evidence to that determination. Although it is out of the scope for this project, we recommend that further analysis should be performed to find more fruitful classes with better results.

Observations

The existence of viral videos, one-off successes, and longtime YouTubers with strong support shed light on the different possible kinds of content creators. If there are observable patterns that relate the number of videos uploaded, subscribership, and total viewership, then further analysis could be possible. Figure 6 shows the relationship between the number of subscribers and the total views of channels in our dataset.

Figure 6: Subscribers and Total Views



We see here a relatively linear relationship. This is to be expected, as most subscribers follow their favorite channels with the purpose of watching their videos.

Figure 7: Subscribers and Video Uploads

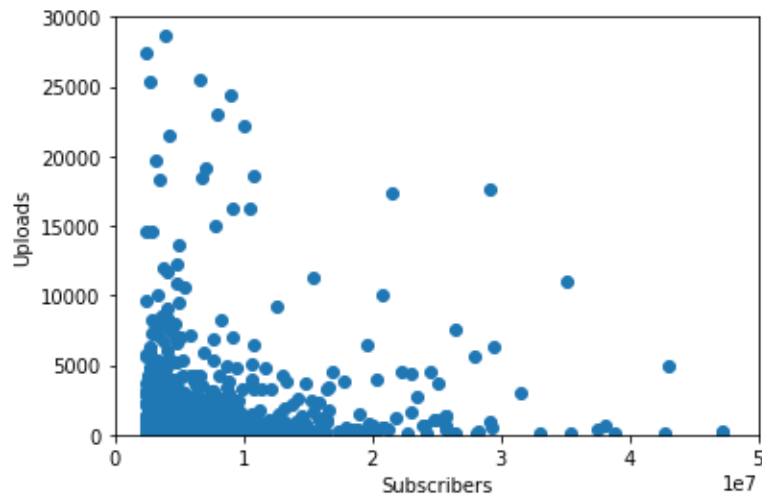
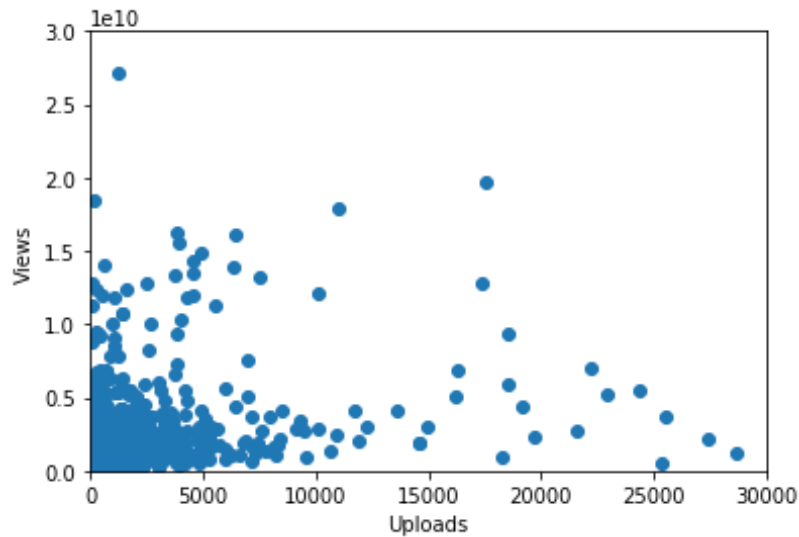


Figure 7 shows a more interesting pattern. There are channels with thousands of uploads and comparatively few subscribers. In fact, the channels with the most subscribers do not appear to have the most videos. A possible explanation could be a “quality over quantity” mentality for subscribers. A user may be more likely to subscribe with fewer, higher quality videos.

Figure 8: Uploads and Total Views



In Figure 8, somewhat similar conclusions can be made as in Figure 7. The large cluster near the origin corresponds to the more common smaller YouTubers with few uploads and few total views. A rough negative linear correlation can be seen between the number of uploads and total views. From this, it may seem worthwhile for a creator to concentrate their efforts on fewer, more interesting videos rather than many videos.

Conclusion

Our findings have shown promise in our data using Regression. It is possible to make conclusions about a YouTuber's total viewership, and therefore success, by looking at their social media presence and YouTube behavior. Classification proved to be a more difficult task, largely due to the small number of channels in the medium to largest viewership classes. Further analysis needs to be done to determine other interesting classifications within our data.

Among these difficulties, we did observe interesting patterns within YouTube behavior itself. We interpreted these as the preferences of views and subscribers towards the uploading habits of content creators. Our conclusion is that viewers and subscribers show a clear preference for fewer videos rather than many. The correlating factor could more time spent to produce more appealing videos in place of low quality, high volume channels.

We provide our dataset as the main feature of this project. The analysis in this paper demonstrates its potential, and to our knowledge, it is the first of its kind.