Popularity Analysis: Factors Influencing Video Engagement Among Top YouTube Creators

Mason Hu

May 1, 2024

Abstract

After an end-to-end YouTube web-scraping session in the midterm with exploratory data analysis, I am ready to carry out actual analysis, fit models, and conduct deep learning on the dataset that is cleaned. I also took into account the feedback Prof. Franklin gave after the midterm.

In this project, I made five interactive plots—3D scatterplots, overlaid histograms and faceted boxplots, and geographic leaflets—and deployed them in a website using Python, Flask, Bootstrap 4, and Railway hosting. I also conducted t-tests after verifying the assumptions, and I found that the publish day of the week of a video is a significant indicator of video views. Moreover, I fitted decision tree models and resulted in a 0.09 MSE. Furthermore, through deep learning via convolutional neural networks, I attempted to learn and predict video categories given video thumbnails image data, for the purpose of predicting video views through CNN-derived video category as input.

See launched data dashboard website for summary. The code and dataset are published on my GitHub.

Contents

1 Introduction

YouTube has been a platform we cannot live without. Whether we want to follow the sports news or reinforce a concept you were taught in class, or perhaps even watch some recreational skits, we can't manage without YouTube. Many many people has made a living doing YouTube, and today we are going to focus on those who are exceptionally successful on this indispensable platform.

Among the top 1000 YouTube giants, YouTubers have videos with high view counts and low view counts. In this midterm data analysis project, I will determine why some videos created by popular YouTubers with an absurd amount of subscribers sometimes still get low view counts, and what in general are the specific factors that drive viewers to view videos.

My research questions:

- What factors lead to high video view count?
- What in turn lead to low view count, despite the YouTuber having high subscribers count?

To answer this question, I turned to Kaggle for datasets. However, the datasets there are scarce, and most of them only has the basic columns with no insightful attributes. Instead, I used only one column of a Kaggle dataset—the links to the channels of top 1000 YouTubers—and started my web scraping journey towards two other huge datasets that I extracted from the web. Guess what, it was not easy.

2 Methodology:

2.1 Data acquiring:

Details of the web-scraping was discussed in the midterm report.

2.1.1 Other wrangling:

1. **Log Transformation for Numerical Variables:** I applied a logarithmic-base-10 transformation to the numerical variables in both the `channels` and `videos` datasets to normalize their distribution and reduce the effect of outliers.

2. **Removal of Irrelevant Data:** In the `channels` dataset, I removed entries that do not meet the following criteria:
    - Channels must have videos; channels with zero videos are removed.
    - Channels must have views; channels with zero views are removed.
    - Channels must have at least 10 million subscribers.
3. **Date and Time Transformations:** I converted datetime fields into more granular components, such as days and hours, to facilitate time-based analysis in the `videos` dataset.
4. **Category ID to Name Conversion:** I mapped numerical category IDs to their corresponding category names for better readability and analysis in the `videos` dataset.
5. **Duration Categorization:** I categorized the duration of videos into three classes—short, medium, and long—based on predefined thresholds to simplify the analysis of video length effects.

These steps are instrumental in preparing the datasets for any subsequent data exploration and analysis, ensuring the reliability and quality of insights derived from the data.

By Prof. Franklin's advice, I included more univariate visualizations here.

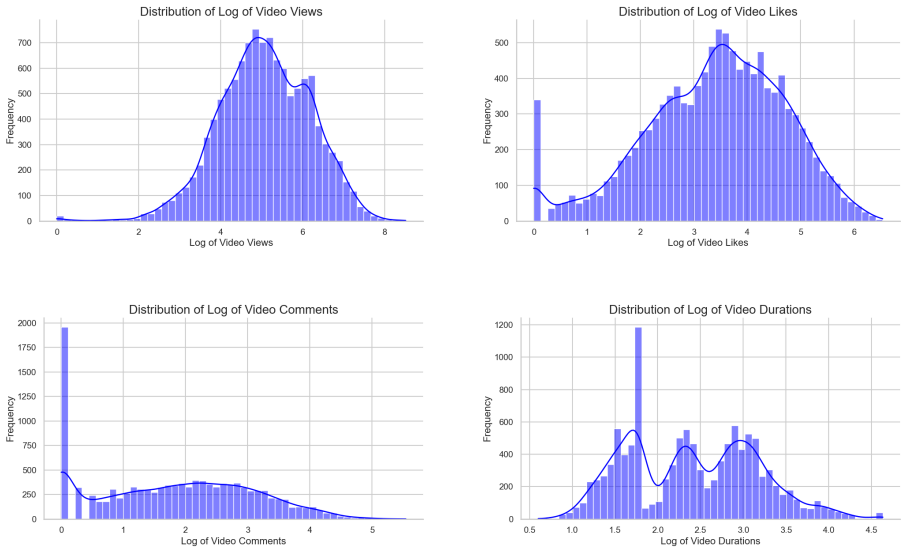


Figure 1. Histograms for the four numerical variables after log transform

What is noticeable is that the logarithm of video durations in terms of seconds is trimodal, which is why I decided to categorize the video duration lengths into three discrete categories. I used the cutoffs obvious in the histogram: [0, 1.9, 2.7, 5]. Those cutoffs correspond to 60 seconds (1 minute) and 600 seconds (10 minutes) which make sense in YouTube videos since a lot of videos shoot for these thresholds.

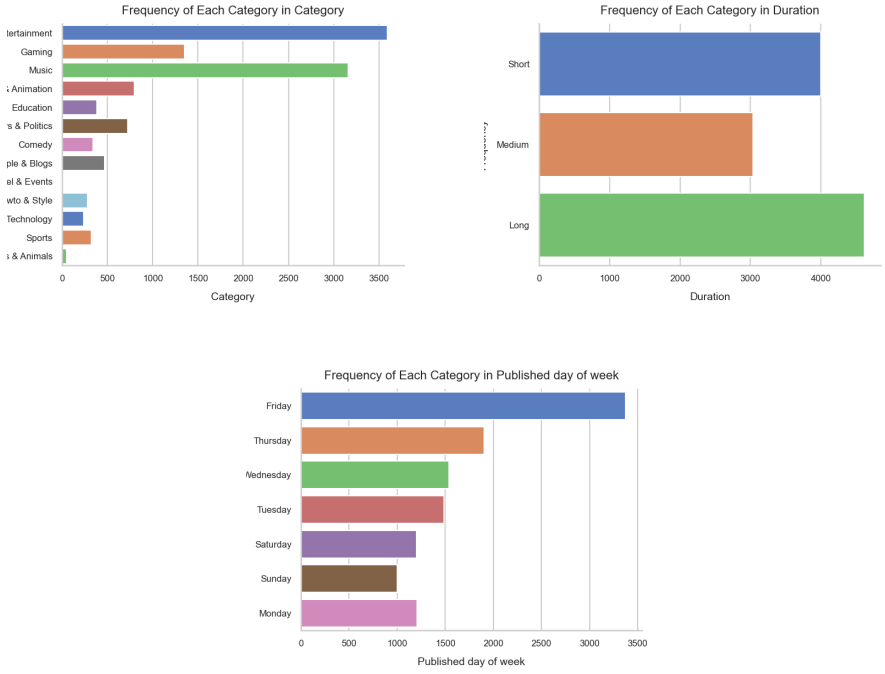


Figure 2. Barplots for the three categorical variables

2.2 Data visualization tools

In my analysis and visualization of video and channel data, I've created a series of figures using `plotly.express` and `plotly.graph_objects` to facilitate an interactive and deep exploration. Each figure is designed to illuminate different facets of the data, as explained below:

2.2.1 3D Scatterplots:

I use two 3D scatter plots to separately investigate relationships in the two large-scale datasets. Figure 1(1) explores the dynamic between channel subscribers, video counts, and total views, aiming to understand the impact of channel content volume on popularity. Figure 1(2) delves into the interplay between comments, likes, and views, providing insights into viewer engagement and content reception. I chose logarithmic scales to manage skewness and enhance readability, using color coding to highlight the intensity of viewer engagement effectively.

2.2.2 Overlaid Histogram:

I created a histogram to visualize the distribution of video views across different video durations and caption statuses. I employed different colors to distinguish between the presence or absence of captions.

2.2.3 Faceted Boxplots:

The boxplots grouped by day of the week and categorized by content types such as news, music, and gaming allows for a direct comparison of views, facilitating the identification of weekly viewing patterns. I selected custom colors to clearly differentiate between work-related and entertainment-related categories, enhancing the plot's clarity and the insightfulness.

2.2.4 Geographic Leaflet:

My geospatial visualization maps channels on a world map with sizes proportional to views and colors indicating different topics. This design decision helps highlight geographical trends in channel popularity and integrates detailed interaction data for a comprehensive overview. I ensured that each data point offers a breakdown of views on hover, enriching the user's interaction with the map.

The proudest part of this visualization is that I incorporated `pycountry` and `countryinfo` libraries to convert alpha2 country codes to alpha3 country codes and more importantly, since YouTubers do not have an exact latitude and longitude coordinate, I manually centered YouTubers on their countries centroid and **added random Gaussian noise with standard deviation scaled linearly by the square root of their country size** on their actual longtitudes and latitudes, so that they look spread out instead of clustered on the centroids.

The resulting map looks great!

Each figure is carefully designed to highlight different dimensions of the dataset. The actual figures are shown in the next section.

2.3 Statistical Models and Deep Learning

2.3.1 T-test

I conducted reliable statistical test to understand the impact the day of the week on which videos are published, the length of the videos, and whether the videos have captions. For each of these categories, the variable of interest is logged views, which serves as a proxy for viewer engagement.

I also performed several statistical tests to validate the assumptions necessary for accurate analysis and to derive meaningful conclusions. Initially, the Shapiro-Wilk test is employed to check the normality of the distribution of logged views, which is crucial for the validity of subsequent t-tests. Following this, Levenne's test is applied to examine the equality of variances between groups, a prerequisite for conducting t-tests under the assumption of equal variances. Finally, t-tests are conducted to determine if there are statistically significant differences in the mean logged views between different categories. This methodical approach ensures the robustness of the analysis.

2.3.2 Random Forest

I preprocessed a dataset focusing on relevant features such as categoryName, publishDay, and viewer interactions like likes and comments. I utilized one-hot encoding to transform categorical variables for compatibility with machine learning models. I chose the Random Forest regressor for its ability to handle both numerical and categorical data effectively and to mitigate overfitting through its ensemble approach and trained the model, evaluating its performance using the Mean Squared Error (MSE) metric.

2.3.3 Convolutional Neural Network

I employed a deep learning approach using convolutional neural networks (CNNs), specifically leveraging a pre-trained ResNet18 model, to predict video categories based on thumbnail images. The primary objective was to ascertain the potential of CNN-extracted features from thumbnails in predicting video views indirectly by first accurately categorizing videos. The method involved fine-tuning

the ResNet18 architecture to fit the classification task by adjusting its final layer to output predictions across the range of unique video categories in the dataset.

I made the custom PyTorch dataset by downloading the thumbnails through URLs using HTTP request responses. I trained the model using batch size 4 for 10 epochs. This approach demonstrates how deep learning can be applied to media content analysis, specifically in automating the understanding of visual content to enhance viewer engagement predictions.

3 Preliminary Results

3.1 Visual Results

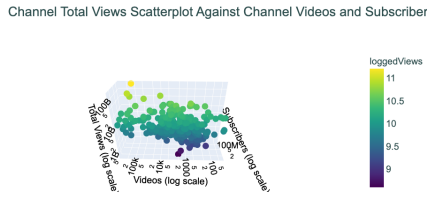
Below are the interactive visualizations outlined in section 2.2. For full visualizations, please visit the data dashboard. As we can see in the Figure 1(1), a channels total videos views are definitely log-linearly positively correlated with the channel's subscribers and video count. Rotating the axes we can visualize the marginal effects of each factor on video views and they all seem log-linear.

As we can see in Figure 1(2), a videos total views are absolutely log-linearly positively correlated with the video's comment count and like count. The correlation is strong and there seems to be a dominant linear relationship in the log scale. Silimarly, rotating the axes we can visualize the marginal effects of each factor on video views and they both seem log-linear.

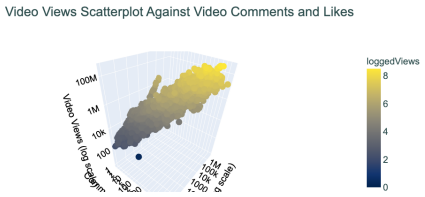
As we can see in Figure 2, the video views are approximately normally distributed for each of the levels and combination of the levels of durationCategory and caption, the medium length videos seems to be more variable in terms of view engagement and the other two categories have a bigger spike in their modes. We can also see that the long videos with captions tend to be have more views on average.

From midterm we knew the average distribution of video views by category. As we can see in Figure 3, a video's view is definitely influenced by the video's category and its publish day of the week and category. In general, education and gaming videos yields way more views than other types of videos in the boxplot. However, we see that specifically, the entertainment videos and new&politics videos has a hard drop on Fridays. film&animations are also lower on Fridays than usual. Moreover, there seems to be barely any news&politics videos on Saturdays and Sundays. Other groups of videos tend to stay consistent in terms of views throughout the week, but they all exhibit signs of diminishing views on Fridays.

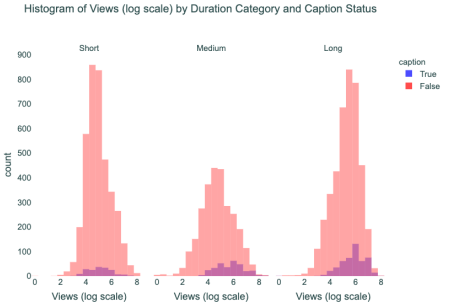
As we can see in Figure 4, the geospatial distribution of channels with their



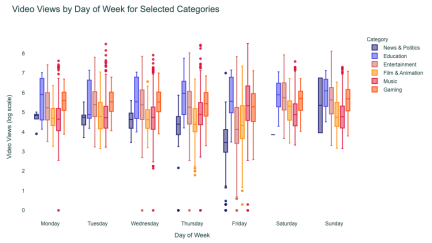
(a) Figure 1(1)



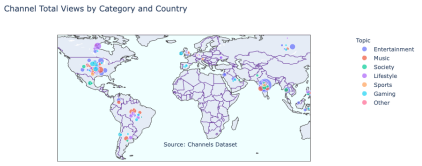
(b) Figure 1(2)



(c) Figure 2



(d) Figure 3



(e) Figure 4

Figure 3. Interactive Visualization posted on the data dashboard

subscriber counts is distinct. Most YouTubers are located in the US and they mostly have high subscriber counts. Surprisingly, there seems to be more and larger YouTubers in India than there is in Europe. There are also many YouTubers in Brazil. There are no YouTubers shown in the map in Africa and China.

3.2 Test results

The following tables are p-values for a series of robust statistical tests for analysis of variation of video views affected by three different video features. The Shapiro-Wilks test is the normality test, with p-value less than 0.05 indicating violation of normality. Levenne's test is for homoskedasticity, with p-values less than 0.05 signalling violation of equal variance. And the t-test showing if there are difference between mean logged views of the given category and other categories in the table.

Table 1. P-values for Statistical Tests for Logged Views by Day of the Week

Day	Shapiro-Wilks	Levene's Test	T-Test
Monday	1.408×10^{-11}	2.311×10^{-13}	5.440×10^{-05}
Tuesday	3.648×10^{-11}	2.810×10^{-17}	1.656×10^{-08}
Wednesday	3.144×10^{-09}	5.671×10^{-11}	2.420×10^{-02}
Thursday	1.295×10^{-12}	7.766×10^{-07}	2.920×10^{-02}
Friday	5.353×10^{-13}	1.606×10^{-129}	1.580×10^{-147}
Saturday	2.479×10^{-05}	9.622×10^{-18}	5.866×10^{-45}
Sunday	2.261×10^{-08}	1.112×10^{-12}	3.912×10^{-21}

Table one shows that although there are violations to the assumption tests, every level in the factor 'day of the week' influences video views substantially. However, the influence of a video being publish on Friday is especially emphasized since the p-value is on a another significance level than the other six.

Table 2. P-values for Statistical Tests for Logged Views by Video Length

Length	Shapiro-Wilks	Levene's Test	T-Test
Short	3.586×10^{-21}	4.919×10^{-45}	7.230×10^{-11}
Medium	7.248×10^{-08}	4.958×10^{-43}	5.256×10^{-38}
Long	6.249×10^{-26}	3.172×10^{-05}	7.403×10^{-86}

Table 2 shows evidence for video lengths being a significant impact on video views, even though the assumptions fail to verify.

Table 3 shows evidence for caption status being a significant factor of video

Table 3. Statistical Tests for Logged Views by Caption Status

Caption Status	Shapiro-Wilks	Levene's Test	T-Test
With Caption	3.671×10^{-07}	2.625×10^{-03}	1.260×10^{-66}
Without Caption	1.822×10^{-24}	2.625×10^{-03}	1.260×10^{-66}

views. Notice the two p-values are exactly the same due to the nature of a 2-level t-test.

3.3 Model Results

3.3.1 Random Forest Result

After hyperparameter tuning, the optimal random forest ended up with 0.093 mean square error.

3.3.2 Deep Learning

I expected this to be the pinnacle of the project. However, due to HTTP requests being too slow and other time constraints (discussed in the section below) it did not end up producing significant results.

4 Summary

As summary of previous results, the video views among top 100 YouTubers are positively related to their subscriber count, their video counts, comments, likes, caption status. The categories, location, length of the video, and the day of week on which they published the video also influences the video engagement they yield.

YouTube API ended up being restricted in a sense. However, they opened up brand new potential that we never thought we had. This is why I devoted two sections to discuss about the limitations and advantages of my approach.

4.1 Difficulties and Limitations

4.1.1 Website deployment in Python

As I communicated with Prof. Franklin ten days ago, I indeed made a unfortunately bad choice doing the midterm in Python. GitHub does not support deployment of potentially non-static websites like those hosted in Python. This left me with three choices: I could either screenshot all my results and use markdown format to produce a GitHub website. However, this would sacrifice interactiveness, which we don't want. I could also transport all my code to R-Studio, and follow the standard procedures discussed in the labs. However, I am not even certain if R supports YouTube data API and even so, it would be tedious and it just sounded like an uncomfortable way of dealing with the situation. I ended up exploring the third option, which is using Flask and HTML, CSS to make a website.

However, that left me with whole lots of other trouble other than merely learning how to make a website in Flask. I have not learned in U of T how to deploy a website after making it locally. So I have to also find a hosting service. In the end, after endless searching for free plans, I chose Railway hosting, and successfully deployed the website in Python.

4.1.2 Scraping problem with IDs

It was mentioned in the midterm report that a call to the search method of YouTube data API has a quota cost of 100 units instead of 1 as other calls do, which uses up out the daily limit (10,000) in no time.

I started early this time to try to spread my search method costs to several days. All this is to obtain better and more complete data collection. However, the YouTube API has bugs with finding channels by IDs (maybe they restricted this intentionally) and I still only end up getting 244 channels out of 1000, which leaves us with our biggest data drawbacks.

4.1.3 HTTP requests

As mentioned before the ResNet18s did not end up performing. This is mainly because a single access of the thumbnails through URLs costs me about 0.3 seconds. Aggregated to the whole dataset of 12 thousand videos, a whole iteration through the dataset (without any computation done), would cost me exactly an hour. Think of how many iterations of data we have to go through in each training session, that is $2 \times num\_epoch$ training and evaluation for each epoch,

which means we have to spend about 20 hours + the mult-adds in the forward and backward passes. What is worse about this is that they we can't really speed the HTTP request up with GPUs.

Therefore, the new plan is that after the final project is due, I will keep carrying out this analysis by downloading the thumbnails first locally (or externally on a drive since the data is big), and train in real time.

4.2 Future Work:

4.2.1 CNN-derived category on the effects of ML models

As mentioned before, the aim of the category learning project is to predict video categories from thumbnails and ultimately, compare the performance of my random forest models (or xgb, glm potentially) using actual ground truth category as input against those using CNN predictions of video categories as input. The performance would be expected to degrade. Therefore, the lack of deterioration in performance would suggest significance of this research. (There is a chance this happens because the video categories are not strict—a gaming video could be valuable as a predictor for video views if categorized as a entertainment video.)

In addition to the ongoing category learning project, I can also further investigate the effects of downsampling on video category classification, since the strong YouTuber API provided us with different resolutions of thumbnails.

4.2.2 Using transformers to investigate topics and/or descriptions

One other thing we did not use are the topic urls. They link to the wikipedia page of topics of interest. This might be super meta, but what if we can predict based on the wikipedia page. This, including further analysis of video/channel descriptions are highly potential future directions worth giving a try.

4.2.3 Integrating SQL

There are tons of meta information like comments, lists of tags and supertags. What if we can expand the database and fully connect every information of a youtube video. This cannot be done without SQL and a big chunk of the rest of my semester. However, if we can accomplish that, it would be huge for making categorical analysis and decision tree models.

References

Kaggle dataset: linked

YouTube data API V3 linked

Appendix

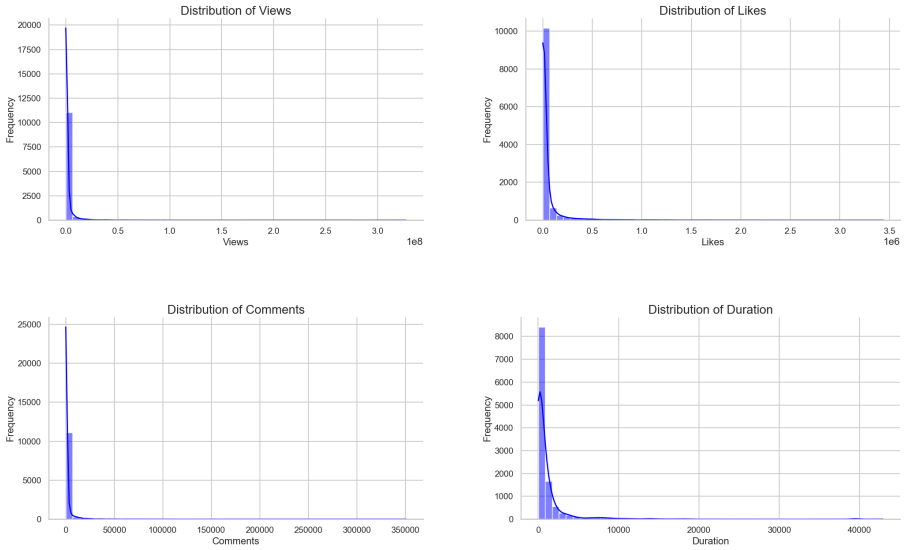


Figure 4. Histograms for the four numerical variables before log transform