
Predicting YouTube Video Views from Video and Thumbnail Features

Lean Ting Jin

Matrikelnummer 6956985
dummymail1@uni-tuebingen.de

Finn Springorum

Matrikelnummer 6124977
finn.springorum@student.uni-tuebingen.de

Christian Traxler

Matrikelnummer 6969273
christian.traxler@student.uni-tuebingen.de

Anna Chechenina

Matrikelnummer 6987499
dummymail1@uni-tuebingen.de

Abstract

We are planning to use the YouTube Data API v3 [1] to see which factors of a YouTube video best predict how many videos it will receive.

1 Introduction

The YouTube algorithm is notorious for its enigmatic and capricious nature, which often results in the arbitrary promotion of specific videos. However, it is ultimately the user who decides in a split second whether a video is being watched. Content creators, especially those in the entertainment sector, aim to maximize their success in terms of video performance, perhaps even live from YouTube, and may have a team with a thorough viewer engagement strategy. In this context, the thumbnail, a fundamental component of the video's presentation, plays a pivotal role in influencing user engagement and selection, as indicated by several studies. However, it remains unclear whether eye-catching thumbnails really lead to a higher view count on average and which aspects of an image might be dominating its interestingness. The present study aims to examine the relationship between diverse image features that are hypothesized to correlate with visual appeal and the view count of the corresponding entertainment video.

2 Methods

Data Collection. The YouTube Data API [1] is a powerful and convenient resource for research pertaining to YouTube video and channel statistics. Its highly parameterized list search method enables the input of a specific query and the acquisition of detailed, publicly visible features for hundreds of videos. Given our primary interest in videos intended for entertainment, designed to appear in the feed of a user and elicit a response, our focus did not extend to videos explicitly sought after through searches (e.g., Film & Animation), videos which are unlikely to include human faces on their thumbnail (e.g., Pets & Animals), or videos from a very niche field (e.g., Nonprofits & Activism). Therefore, we decided to include eight of the 15 main YouTube video categories, namely Comedy, Education, Entertainment, Gaming, How-to & Style, News & Politics, People and Blogs, and Sports. Due to the limited number of videos that can be retrieved for a given query, the date range from January 1st, 2015 to the time of collection was divided into disjoint sets, and 500 videos were requested for each period. This strategy enabled the acquisition of thousands of videos for a single query. To prevent duplicates while respecting the daily API call limit for a user, our data collection strategy entailed the request of up to 2.5k videos from a designated category, each accompanied by a singular generic keyword (e.g., most popular video games for the Gaming category), until the collection comprised precisely 10k unique videos from said category, yielding a dataset with a total

of 80k data points.

Video and Thumbnail Features. For each video, a multitude of characteristics were saved, most notably the view count, subscriber count, and the URL to the high-resolution version of its thumbnail. Subsequently, a pipeline was implemented that temporarily downloaded the thumbnails and computed the requested features like the average hue (in radians), saturation (normalized to [0,1]), and lightness (normalized to [0,1]) across all pixels, as well as the contrast (standard deviation of the normalized grayscale image) and sharpness (variance of the Laplacian of the grayscale image). We further utilized the DeepFace library [2, 3] in high-accuracy "retinaface" mode to extract the predicted number of human faces in the thumbnail.

Analysis. We predominantly employed linear regression methods due to the implausibility of more intricate relationships between the view count and the one-dimensional eye-catchiness features. However, since linear regression methods require the residuals to be homoscedastic and normally distributed, we first log-transformed the view count whose variance substantially increases with the order of magnitude. In addition, we introduced a critical step in our analysis: Since the YouTube algorithm clearly favors videos from channels with lots of subscribers, which is expressed in their view count, we normalized the view count based on the respective subscriber count. Specifically, we employed a linear regression of the log-transformed view count against the log-transformed subscriber count, subsequently utilizing the corresponding residuals. This preprocessing approach was designed to elucidate the fluctuations in view count that were not explained by the subscriber count, but potentially by the different thumbnail features that were considered.

3 Results

4 Discussion/Limitations

Discussion

Limitations As a project based on the YouTube API, there are couple of inherent limitations of the data collection. One issue with the data collection is the categories: each video is either user-defined or automatically assigned only one category. This means that there exists some error in the categories such that errors are likely and that some categories that we wanted to exclude (like music videos) are likely to be included and may skew results.

Also,

5 Statement of Contributions

Here is an example:

XX performed the correlation analysis, organized the data and code for the processing of dataset1 and subdataset2, and created the scatter plot. YY created the random forest regression model, performed the data cleaning for the xyz analysis / xyz database, and created the bar charts to display the regression results. ZZ researched and collected the raw data, restructured the pipeline for the data analysis, and proof-read the draft for the final report. AA performed the data cleaning for dataset1, and performed the Ridge and Lasso regularization. All members of the group contributed to writing the report.

References

- [1] Google. YouTube Data API v3. <https://developers.google.com/youtube/v3/docs>.
- [2] Sefik Ilkin Serengil and Alper Ozpinar. A benchmark of facial recognition pipelines and co-usability performances of modules. *Bilisim Teknolojileri Dergisi*, 17(2):95–107, 2024.
- [3] Sefik Ilkin Serengil and Alper Ozpinar. Lightface: A hybrid deep face recognition framework. In *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pages 23–27. IEEE, 2020.