# An Analysis of YouTube Trending Videos

Jeremy Wang (jw2363) and Lorenzo Scotto di Vettimo (ls769)

December 5th, 2021

## Introduction

While videos being recommended to users are effective for more watch-time, YouTube Trending page has increasingly become a more competitive edge because videos compete for a national (country-level) ranking every millisecond. And, these videos are not trivially just the most viewed YouTube videos of some time period. The added complexity makes the Trending page mysterious yet interesting enough for viewers to interact with the videos by watching them, liking/disliking them, commenting on them, and sharing them. Since the viewers of YouTube value the Trending page, many YouTube channels are motivated to put more effort in their videos in order to achieve this status of "trending" and see how high they can climb the trending ladder and how long they can stay on it. The application of being able to gain insights into such a dataset as the Trending page would allow for increase user retention.

## Problem Statement

Our problem statement considers asking the following questions:

- Is a video still on Trending?
- How many days, starting from the uploaded date, does it take for a video to get on Trending?

Notice that both of these questions have an emphasis on either the video already has been on Trending (former question) or the video will be on Trending at some point (latter question). We answer these questions in our Analysis I and Analysis II sections, respectfully. So, we want to look at data from trending videos which we thoroughly discuss in the next section.

### Exploring the Data

#### Overview

The dataset consists of the top 200 videos on Trending per day. We choose to focus our study to daily trending videos in the United States to narrow our focus of analysis. Limiting the region is reasonable because anyone using YouTube in the United States can only look at the Trending page for United States on either YouTube's website or app, and each region has their own YouTube Trending

that does not interfere with any other region's YouTube Trending. So, we will only use the dataset consisting of the USA Trending page videos. This sums up to 89,791 videos with 16 features.

### Original Features

**Text Features**
Feature 1. Video_id - Video ID associated with video that can be used to access link to video
Feature 2. Title - Name of the YouTube video posted when the video became trending
Feature 4. ChannelID - ChannelID associated with the video uploader
Feature 5. ChannelTitle - Name of video uploader channel
Feature 8. Tags - Tags the video uploader decides to use for the video
Feature 13. Thumbnail_link - A link to a picture of the thumbnail for the video
Feature 16. Description - The description of the video written by the uploader
**Numerical Features**
Feature 9. View_count - The number of views the video has once reached trending
Feature 10. Likes - The number of likes the video has once reached trending
Feature 11. Dislikes - The number of dislikes the video has once reached trending
Feature 12. Comment_count - The number of comments the video has once reached trending
**Boolean Features**
Feature 14. Comments_disabled - A boolean value to indicate if comments were disabled
Feature 15. Ratings_disabled - A boolean value to indicate if the ratings were disabled
**Continuous Features**
Feature 3. PublishedAt - Date and time when video was uploaded to YouTube
Feature 7. Trending_Date - Date when video reached "Trending" status on YouTube
**Categorical Features**
Feature 6. CategoryID - Integer used to refer to one of 31 possible categories of videos

Figure 1: Features of the original dataset grouped by data type.

In Figure 1, we have the original 16 features from the dataset downloaded from Kaggle. We have important features like the number of views, number of likes, number of dislikes. Notice that all these features are the values at the time when the top 200 trending videos were extracted from YouTube and added to the dataset on Kaggle. The time of extraction is always the same at midnight for UTC timezone which equivalently means midnight in Greenwich, London. More importantly, what this means for certain features like the *Comment_count* column, it consists of the number of comments while it was trending at midnight in London's time, and for *Likes*, it is the number of likes the video has while it was trending at midnight in London's time. So, our biggest assumption for our data and predictions is that we assume a video stays on the Trending page only by day and not by a more precise unit of time because our data is limited to only extracting Trending videos once each day. So, instead of using the number of likes at the time it was trending exactly, we can only use the number of likes at the time of extraction. Therefore, we predict by the days unit of time and nothing more precise. But generally the most useful unit of time for asking our questions is by day, so it is reasonable to continue with this data and assumption.

In Figure 2, we show the distribution of our only categorical feature, which is the category of a video. Examples of categories are Music, Sports, Gaming, Comedy, Entertainment, Pets and Animals, and Science and Technology. Since each category is assigned to a different number, the order of them does not matter for our modeling. Videos of any categories are considered in the YouTube Trending page, so we do not remove any particular category. As can be seen for the trending video categories, most of them seem to relate to category number 24 and 10, which are Family and Comedy respectively. This is somewhat logical, as the "Family" category reflects how YouTube features many content creators

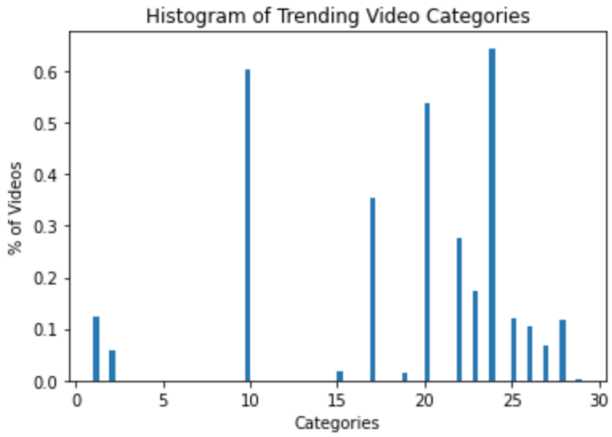


Figure 2: Categories of trending videos on YouTube. The numbers refer to internal categorization of a possible category of YouTube videos

encourage user engagement through vlog style videos. As they want watchers to become friendly and think of the creator as a welcoming friend, the popularity of the "Family" category seems to reflect this. The "Comedy" category also seems logical with the popularity of many videos to provide entertainment and laughter to the audience. Other categories that stand out as having the most videos on trending are Music, Sports, Gaming, and Entertainment.

**Examples of Data**

Here are four videos that were extracted as you can see by the time it was extracted in *trending_date* which we use as the actual trending time based on the assumption described above. Note that not all the data is from at the time of extraction. For example, the *publishedAt* column is the actual time it was uploaded so it cannot be changed. Other features like the title can technically be changed at any time, but we assume most YouTube channels do not change that if not just for minor grammar mistakes in the title. This leads us to which features we actually use out of the original ones listed.

**Missing Entries**

First, we check for missing values in our dataset. There are some missing entries, but they only occur in *Description* which means the content creator uploads a video with nothing in the description since it is not required to be filled. This is the case only for 1.5% of the data which is 1,136 videos. Interestingly, it is pretty rare to get a video on Trending with no description at all because in general it is hard to find any popular video (whether on Trending or not) that does not include some sort of description even if it is just a few words.

Now, in our dataset, the first set of Trending videos is extracted from August 12th, 2020. The last set of Trending videos is extracted from October 28th, 2021. However, we have a few days in which there was no extraction of Trending videos at all for that day. Missing in 2020: Oct 28, Nov 22, Dec 6, and Dec 30. Missing in 2021: 2021: Feb 23, Feb 28, May 24-29, Jun 18, and Jun 26. Fortunately, these days are spread out over a few months, so it will not impact our predictions much. We thought of filling these entries in by guessing which videos would be Trending on these days based on the days right before and after these missing entries, but we decided to not include these days for

our dataset since there are too many videos to consider.

And, not surprisingly, there is a decent amount of days in year 2020 (216 days specifically) where no videos were uploaded that made it to top 200 Trending videos. While there are less of these days in year 2021 because we have less months of data, we still have 10 out of the 12 months from 2021 so it is surprising to see a significantly less amount of days (4 days specifically) where no videos that made it to Trending were uploaded. This is why it is crucial to use time as features.

**Feature Engineering**

The times when each video was uploaded and made the YouTube Trending page were extracted from YouTube according to the UTC timezone format, which contains strings and hyphens. We transformed these features to have one column for the upload year, one for the upload month, one for the upload day, one for the upload hour, one for the upload minute, and one for the upload second.

Additionally, we introduced new features and labels better to facilitate the machine learning process. To use within our regression of determining the number of days a video would trend, we computed this supervised label by computing the number of consecutive days a video stays trending. Using this new label, we could used this to compute the average for the number of views, likes, dislikes, and comments across the number of days a video stays trending. These were new features created to be the average number of views per day (the average number of views for all the days a given video remains on the Trending), average number of likes per day, average number of dislikes per day, and average number of comments per day. For our classification problem of computing if a video would still be trending or not, we introduced a binary label to correspond if this day was the last day the video was made the Trending page. If so, this meant that the video would never have made the trending page again and would be labeled "0." However, if the video would continue to trend the following day, the video would then be labeled "1," as the video would still be on the Trending page the day after. Only on the last day a given video makes the Trending page would the feature be labeled a "0."

In our dataset, we used the videos that made the YouTube Trending page in the range from August 12th, 2020 to October 28th, 2021. This gives enough data to cover over a year, which we assumed would be enough to analyze the trending page. YouTube Trending is volatile in that the methods used by YouTube users constantly change in order to improve their chances of their video getting on Trending. So, we could potentially have too much data because videos from 2010 for example were very different in terms of the limited capabilities of technologies used to film and edit them, along with features that YouTube changes over time. Recently, in early November of 2021, a critical policy change by YouTube is that only the uploader is allowed to view the number of dislikes of their video while maintaining the number of likes public. So, we decided to not use more data after they implemented this change because then we would only have barely a month of data from November to December where dislikes are not public, which could greatly affect a video's potential to be trending. From this exploration of the dataset, we had strong reasons to assume we had a sufficient amount of data to utilize.

### Removing Outliers

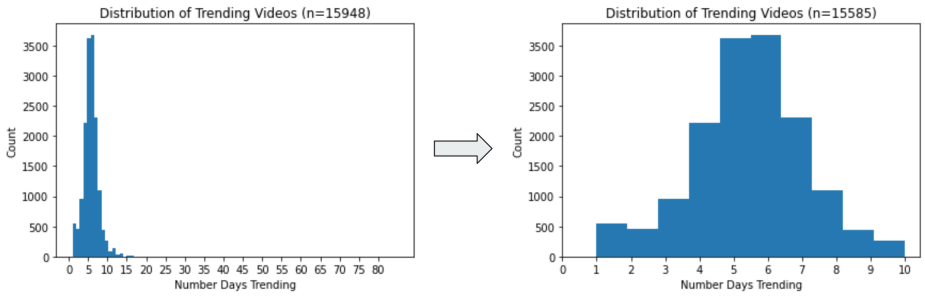


Figure 3: By removing outliers from the dataset, we can see that

As shown in the above figure, When analyzing the number of days videos using exploratory data analysis, we noticed that it could range widely from 1 to 50 days. Upon this realization, we sought to remove outliers by removing videos that trend a significant difference from the median in the dataset, which was determined to be 5 days. This led to a decrease in the total number of videos available in our dataset as shown in the above graph distributions.

## Analysis I: Predicting If a Video Is Still Trending

We now discuss our approach to the first question of determining if a video would still be on the YouTube Trending page the following day. We use supervised learning and treat the case as classification problem using the binary label introduced earlier. A "0" would correspond to the video not making the Trending page the next day while a "1" would.

### Training the Models

When splitting our dataset, we used an 80/10/10 split for training/validation/test split. The features that were used in the model used for a given datum referred to a video on a given day with the number of likes, dislikes, comment count, the number of days the video has been trending, video category, and the number of days it took for the video to be trending since its upload date. As we did not apply natural language processing techniques to our dataset, we did not make significance use of the textual features like a video's title, username of the uploader, video tags, nor the video's description. It is worth noting how because these textual features may be subject by the video uploader at any time point, we avoided a possible confounding effect that may exist with changes that could be made when performing our investigation. As such, our investigation holds merit in studying videos purely based on viewer statistics.

In accessing possible models, we explored three machine learning models: logistic regression, decision trees, and random ensemble using decision trees using the Sklearn library in Python. Hyperparameters were chosen by following a grid search method to ensure that the optimal for the given model. Regularization was also explored during this search for parameters of a model to over overfitting them. For logistic regression, this meant experimenting over the regularization strength $C$, solver, and penalty. For the decision tree model, this meant exploring parameters with the criterion from either "gini" or "entropy", max depth, the minimum number of samples required to split an internal node, and the minimum number of samples required to be at a leaf node. For a

random forest method, this meant exploring the space from number of base estimators used and the maximum number of samples to draw from the training set to train each base estimator. The grid search method was done across the validation set.

### Accuracy of Results

| Model | Training Accuracy | Validation Accuracy | Test Accuracy |
|---|---|---|---|
| Logistic Regression | 82.4% | 82.3% | 81.3% |
| Decision Trees | 82.5% | 82.4% | 82.3% |
| Random Forest Tree | 99.6% | 83.3% | 83.2% |

Table 1: Predicting if a video will be trending the next day.

We report our best performing models along with the training, validation, and testing accuracy achieved for these models in Table 1. As seen, the model which performed the best under the test set was the random forest tree model with an overall test accuracy of 83.2%. While this seemingly high number may suggest our model performed well, the truth of the model can be visualized using a confusion matrix. This useful figure can be used to describe the performance of a classification model on a set of test data for which the true values are known. As each row of the matrix represents the an actual class of the model while each column represents a predicted class. A heatmap is also applied to show where most of the predictions are the highest and closest to 1, which leads to a darker color shown.

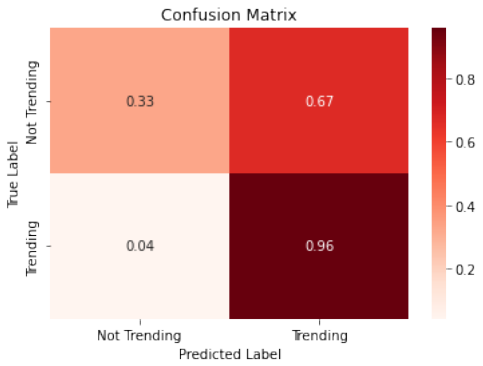


Figure 4: Confusion matrix for visualizing errors in predictions

An investigation into the model performed poorly can be seen in the figure above. The model is able to correctly classify 96% of the videos marked "trending," thus given a label of "1," in the dataset marked. However, the model is only able to correctly classify 33% of the videos marked as "nontrending," thus given a label of "0," in the dataset. Part of this issue is due to an existing class imbalance within the dataset, as 82% of the videos are labeled "1" and the rest labeled with "0." This shows how our model is susceptible to false-positives. This shows how a majority of the videos that are on the YouTube Trending page have already been on the trending page, while only 18% of videos show up once on the page as a one-hit wonder. This highlights an existing imbalance showing that for future work, a loss function to deal with anomaly of "nontrending" videos should be explored for investigating this classification problem.

## Analysis II: Predicting Number of Days for a Video to Get on Trending

We now discuss our approach to the second question to predicting the number days a video would be on the YouTube Trending page. We use supervised learning again but treat the case as a regression problem using the number of days a video would trend label introduced earlier. Because of our removal of outliers mentioned earlier, we may only have videos trend from 1 to 10 days.

### Training the Models

When splitting our dataset, we again used an 80/10/10 split for training/validation/test split. The features that were used in the model used for a given datum referred to a video on a given day with the average number of likes across all days the video is on the Trending page, dislikes across all days the video is on the Trending, comment count across all days the video is on the Trending, video category, and the number of days it took for the video to be trending since its upload date. We again did not apply natural language processing techniques to our dataset for the reasons mentioned in the classification section previously.

In accessing possible models, we explored four machine learning models: ordinary Least Squares (OLS), Ridge, LASSO, and a decision trees regressor using the Sklearn library in Python. Regularization was also explored during this search for parameters for the model to over overfitting the models. Since the LASSO model is the linear model trained with L1 regularization, the hyperparameter $\alpha$ was chosen through a grid search method by experimenting and picking that which yielded the lowest mean squared error (MSE). MSE was chosen as the metric as it is useful in checking how close estimates or forecasts predictions are to the actual values. Since the ridge model is the linear least squares model trained with l2 regularization, the hyperparameter $\alpha$ that yielded the lowest MSE across a grid search was chosen. The grid search method was done across the validation set.

### Accuracy of Results

| Model | Training MSE | Validation MSE | Test MSE |
|---|---|---|---|
| OLS | 3.45 days | 3.36 days | 3.32 days |
| Ridge | 3.46 days | 3.37 days | 3.34 days |
| Lasso | 3.45 days | 3.36 days | 3.32 days |
| Decision Tree | 3.08 days | 3.20 days | 3.28 days |

Table 2: Predicting number days a video will be trending.

We report our best performing models along with the training, validation, and testing MSE achieved for these models in Table 2. As seen, the model which performed the best under the test set was the decision tree model which an overall test MSE of 3.28. One advantage to this regression analysis as opposed to our classification approach is providing a given number to the number of predicted days a video will be trending. However, a possible limitation is our simplification of removing outliers in the data exploration steps. By only considering videos from 1 to 10 in this new dataset,

we model is only good at predicting the number of days a video is trending with a 3 day uncertainty period. In the future, this should be improved upon by future investigators with an exploration of more advanced models such as ControlBurn to find the most important features as well as automated machine learning with Oboe to find the best model.

## Fairness Discussion and Conclusions

Since our classification model predicted a significant amount of false positives, these are harmful in the sense that content creators could develop a wrong notion of which videos get on the Trending page compared to which videos do not. And, since we only use videos that have already been on Trending, this does not give a fair chance to any video that has not been on Trending yet but might be in the future. However, we are limited in data in this regard because any video can technically have the potential to be Trending at some point in the future from any given current time. And, as mentioned before, our modeling only works in a limited time frame from 2020 to 2021 because videos that went Trending before 2020 and after 2021 are more different as time increases away from this range. So, users must be careful when using our models to estimate outside the range of 2020 and 2021.

Our model does not exclude any particular group of content creators based on gender, sex, religion, disability, family status, age, political beliefs, or nationality because none of our features used are correlated with these factors. Specifically, we do not use text features which might indicate factors such as political beliefs in the title or description of a video or might indicate gender based on the YouTube channel's name. A particular sense of fairness in our case is YouTube channels that have one main person as the content creator versus YouTube channels run by corporations. We ensure that neither only individual content creator nor only corporations' YouTube channels are Trending because no feature used can be correlated to this (e.g. not using the amount of subscribers a YouTube channel has).

The way which we described our assumptions implies that there could exist survival bias in the validity of our results by only using videos on the Trending. As a result, we can only make generalizations for videos that make it on the trending page rather than YouTube videos in general as a wide population. At first, we considered predicting in relation to any YouTube video, but after looking at the limited datasets on Kaggle, we realized it would be infeasible since they all only contain videos on Trending already. However, we are able to create a strong baseline model for predicting trending videos that may help content creators and anyone else who is interested in knowing how a trendy a video is.

Throughout this report, we have stated a problem to solve, explored our data, feature engineered the chosen dataset, and finally analyzed the results of models on the dataset along with fairness and limitations. We always seek to help creators and users on YouTube in order to ultimately inspire new creative videos, especially with the rise of shorter and fast-paced videos. We hope that our models can serve as strong initial attempts for content creators in understanding and predicting if their video will continue to trend.

# Bibliography

Dataset used from Kaggle: https://www.kaggle.com/rsrishav/youtube-trending-video-dataset
YouTube's policy change about dislikes: https://blog.youtube/news-and-events/update-to-youtube