Capstone Project 2
Final Report
Content-Based Youtube Recommender System

Problem Statement

Building a content-based recommender system that recommends similar Youtube videos based on the video's features.

Client

Youtube. As someone who admittedly falls into the "Youtube rabbit hole" quite often, I see how effective these recommender systems can be. My goal is to create a simple and efficient recommender system based on the latest data I was able to find.

Data Wrangling and Cleaning

All of the data used in this project is from Kaggle. The following files are going to be used in this project:

- `GB_category_id.json`
- `GBvideos.csv`
- `US_category_id.json`
- `USvideos.csv`

Through data wrangling, we see that this data contains information on the top trending videos in Great Britain and the United States from 11/14/17 to 06/14/18. Each day, there are around 150-200 videos included on that list for each country.

`GBvideos.csv & USvideos.csv`

video_id
trending_date
title
channel_title
category_id
publish_time
tags
views
likes
dislikes
comment_count
thumbnail_link
comments_disabled
ratings_disabled
video_error_or_removed
description

Both csv files contain the columns shown on the left. The variables provide the videos' statistics such as the number of likes and dislikes, number of views, number of comments, and etc.

The `tags` column provides all the tags included in the videos. Video tags/ Youtube tags "are words or phrases used to give YouTube context about a video." Since we would want to see the relationship of the number of tags included in the video and the number of views that video gets, we create a new column called `tag_count`.

The `trending_date` column, which contains the date when that video was on the trending list, currently has an object type. To convert this to a datetime type, we need to change its current format from year/day/month to year/month/day.

Our data for Great Britain contains some null values. Since this is just .23% of the data, we simply drop it. Lastly, we merge both data frames and make it our main data frame. To account for the countries these videos went viral in, we add a column called `country` to indicate whether the data is from GB or US.

`GB_category_id.json & US_category_id.json`

The dataframe of the JSON files look like this:

	kind	etag	items
0	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/1v2mrzYSYG6onNLt2...	{'kind': 'youtube#videoCategory', 'etag': '"m2...
1	youtube#videoCategoryListResponse	"m2yskBQFythfE4irbTIeOgYYfBU/1v2mrzYSYG6onNLt2...	{'kind': 'youtube#videoCategory', 'etag': '"m2...

Since the GBvideos and USvideos dataframes only contain the category_id of each video, we use these JSON files to get the category title that is associated with that id. With that said, we are only interested in the `items` column. Each row in this column contains the following dictionary:

```
{'kind': 'youtube#videoCategory',
 'etag': '"m2yskBQFythfE4irbTIeOgYYfBU/Xy1mB4_yLrHy_BmKmPBggty2mZQ"',
 'id': '1',
 'snippet': {'channelId': 'UCBR8-60-B28hp2BmDPdntcQ',
  'title': 'Film & Animation',
  'assignable': True}}
```

We extract the id and category information from this using a forloop.

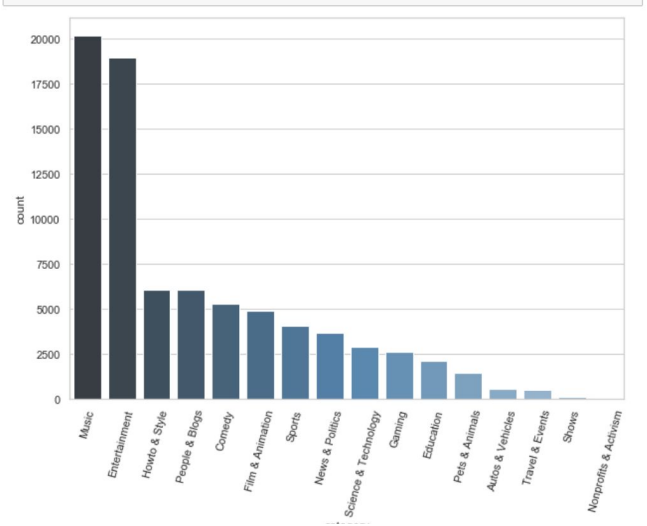
With this information, we add another column to our main dataframe--`category`. This will be one of our key variables in building a recommender system.

There are some null values in the category column but since this is just .23% of the data, we drop it.

Exploratory Data Analysis

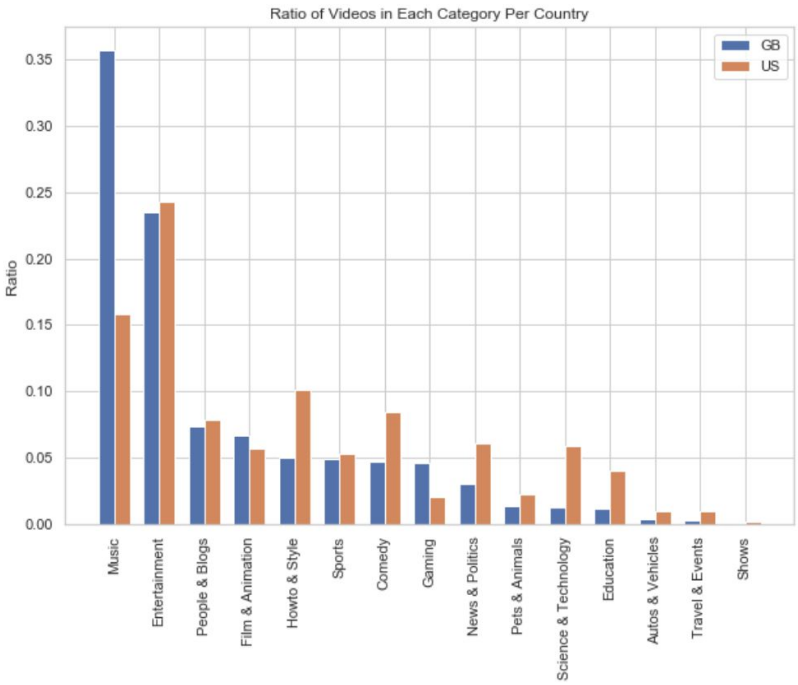
Next, we study trends across the different categories.

First, we see which categories are often on the trending videos list.



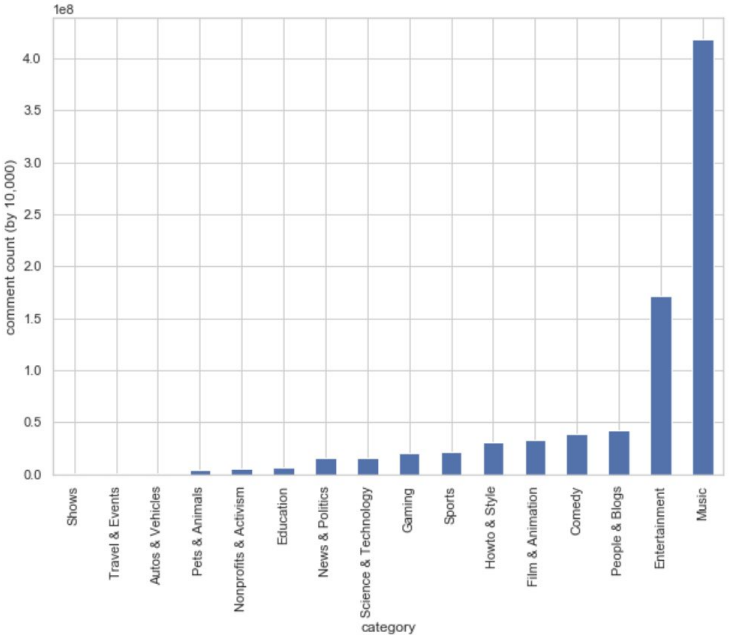
As seen in the count plot, videos that fall in the category of music and entertainment make up for the majority of the trending videos.

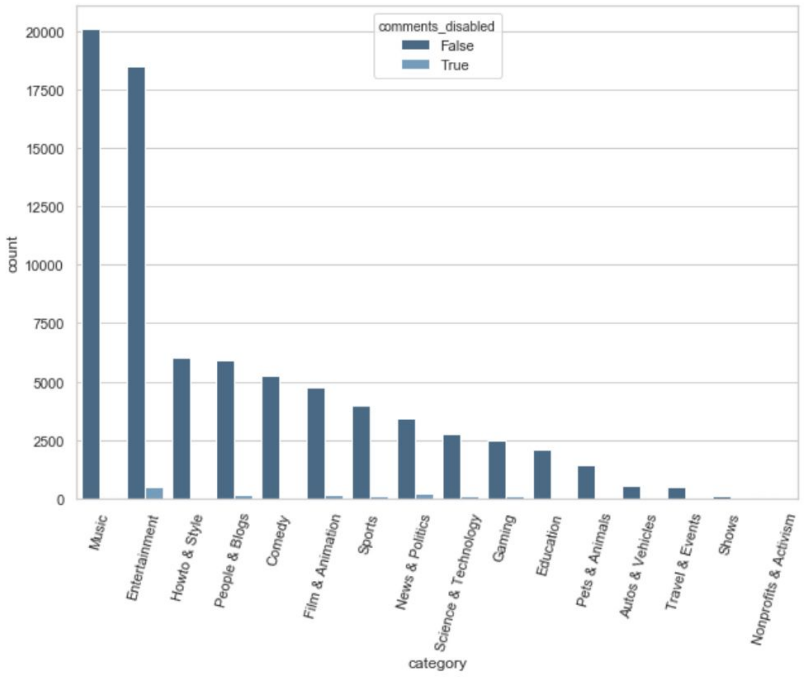
Next, we see if the popularity of these categories is the same in both countries. Since we have more data for US, we cannot simply make a countplot. Instead, we calculate the percentage of trending videos that fall in each category for each country.



As shown in the barplot, the most popular category in Great Britain is Music, while the most popular one in the US is Entertainment.

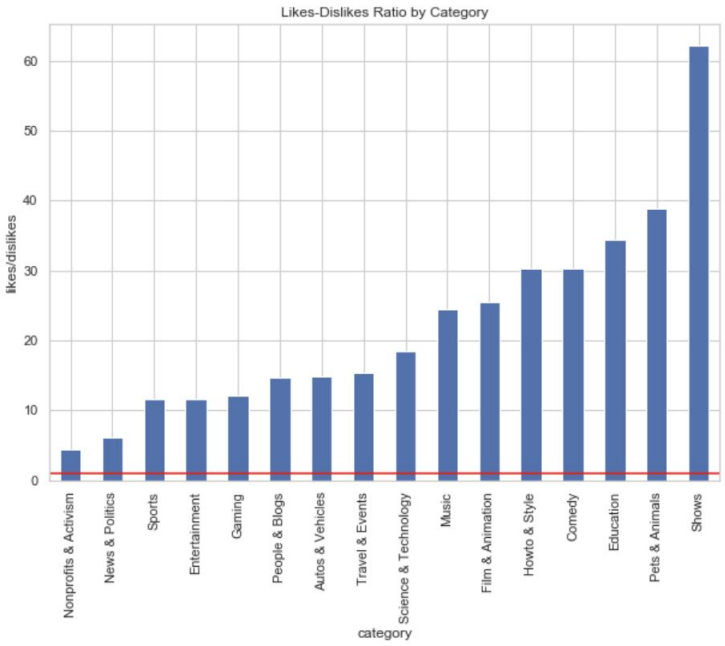
Next, we try to see if comment counts vary across categories. As expected, the most popular categories also happen to have most of the comment counts on Youtube.

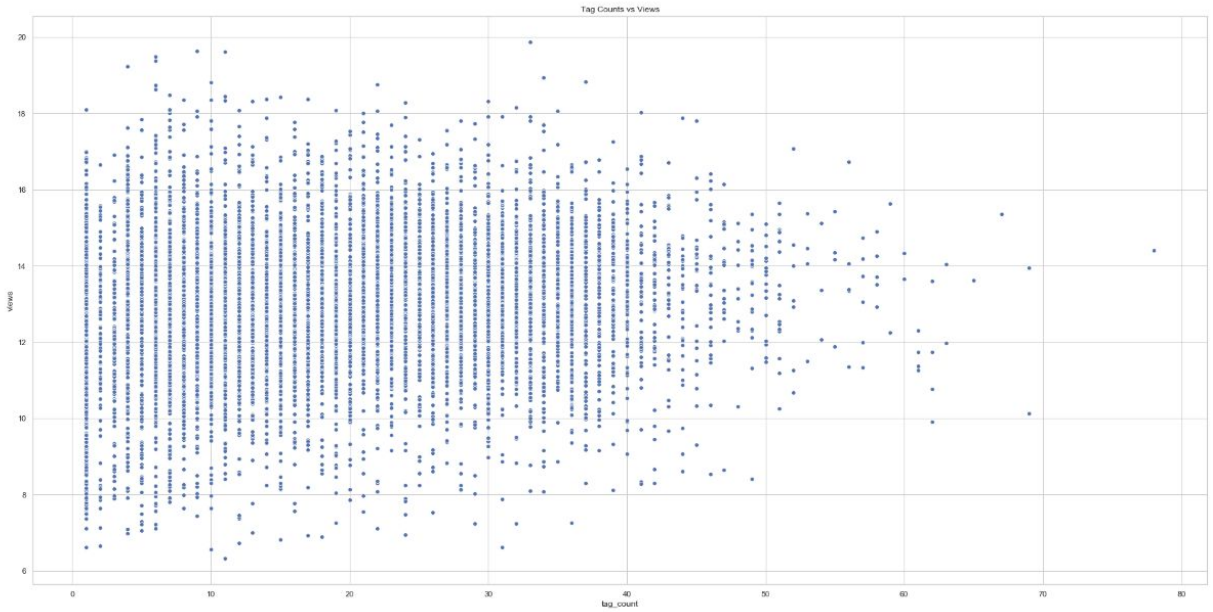




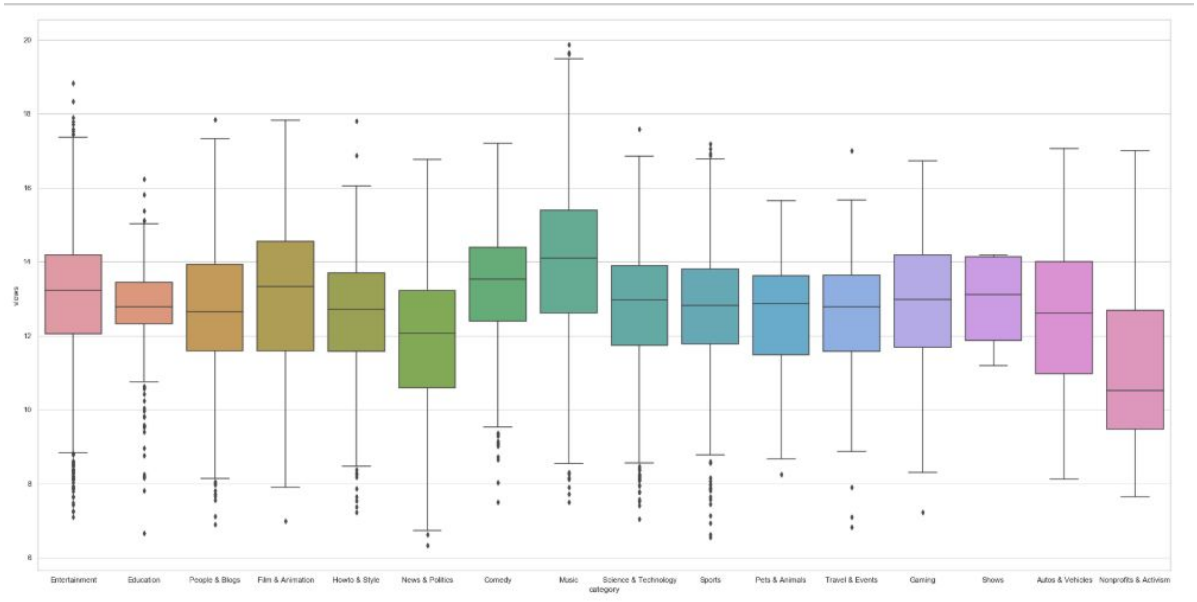
With this plot, we try too see which categories often usually disable comments on their videos. As we can see, most of the categories have very few videos in which comments are disabled. This is helpful information because this shows us that this variable would not explain much about the category.

By getting the likes-dislikes ratio of each category, we see whether or not most of these videos are trending due to positive or negative reviews. A horizontal line is drawn when y=1 (i.e. there are the same amount of likes as there are dislikes. Since no categories fall under the threshold, this means all videos are often well-liked. Although, we can see how close categories such as “Nonprofits & Activism” and “News & Politics” fall closely to the threshold. This makes sense since these videos are often controversial and divisive.





The main purpose of video tags is to help viewers find the video's content. Does this mean videos with more tags often get more views? The scatterplot above tells us it doesn't. The relationship seems random.



In the previous count plots, we see that most of the videos on the trending list belong to the music category. The boxplot above shows that this category also often have the most amount of views. It is important to note that since videos can show up in the trending list more than once, for this plot, we extracted the latest view count for each video.

Next, we try and see which videos showed up on the daily trending list the most between November 2017 to June 2018 in each country. In the US, the most a video has been part of the daily trending list during this time period was 38 times. There were six videos that were on the trending list for 38 days, and three of them belong in the music category.

	video_id	count	title	category
0	NooW_RbfdWI	38	Jurassic World: Fallen Kingdom - Official Trai...	Entertainment
1	ll-an3K9pjg	38	Anne-Marie - 2002 [Official Video]	Music
2	2z3EUY1aXdY	38	Justin Timberlake's FULL Pepsi Super Bowl LII ...	Sports
3	BhIEIO0vaBE	38	To Our Daughter	People & Blogs
4	u_C4onVrr8U	38	Miguel - Come Through and Chill ft. J. Cole, S...	Music
5	u_C4onVrr8U	38	Miguel - Come Through and Chill (Official Vide...	Music

In Great Britain, the most times a video showed up on the trending list is 30. This was only true for one video, which fell in the entertainment category.

	video_id	count	title	category
0	j4KvrAUjn6c	30	WE MADE OUR MOM CRY...HER DREAM CAME TRUE!	Entertainment

Inferential Statistics

Now that we've seen the difference between different categories, we want to see if the differences are statistically significant. Since there are 16 categories that we want to compare, we can use a statistical technique called Analysis of variance (ANOVA). ANOVA uses F-tests to statistically test the equality of means in each group. If the p-value of these F-tests are lower than 5%, we reject the null that the mean across the categories are equal. This will help us determine whether we should include the variable in our predictive model or not.

Views By Category

> Null Hypothesis: The number of views each category gets are not statistically different from each other.
>
> Alternative Hypothesis: The number of views each category gets are statistically different from each other.

```
f-stat: 29.91229
p-value:0.0000000000
The number of views each category gets are statistically different from each other.
```

Likes By Category

Null Hypothesis: The number of likes each category gets are not statistically different from each other.

Alternative Hypothesis: The number of likes each category gets are statistically different from each other.

```
f-stat: 544.81234
p-value:0.0000000000
The number of likes each category gets are statistically different from each other.
```

Dislikes By Category

Null Hypothesis: The number of dislikes each category gets are not statistically different from each other.

Alternative Hypothesis: The number of dislikes each category gets are statistically different from each other.

```
f-stat: 60.05653
p-value:0.0000000000
The number of dislikes each category gets are statistically different from each other.
```

Comment Count By Category

Null Hypothesis: The number of comments each category gets are not statistically different from each other.

Alternative Hypothesis: The number of comments each category gets are statistically different from each other.

```
f-stat: 148.42400
p-value:0.0000000000
The number of comments each category gets are statistically different from each other.
```

Tag Count By Category

> Null Hypothesis: The number of tags the videos in each category are not statistically different from each other.
>
> Alternative Hypothesis: The number of tags the videos in each category are statistically different from each other.

```
f-stat: 51.10537
p-value:0.0000000000
The number of tags the videos in each category has are statistically different from each other.
```

In conclusion, all of these features are statistically different across categories. This tells us that all of these features should be included in our recommender system.

Building the Recommender System

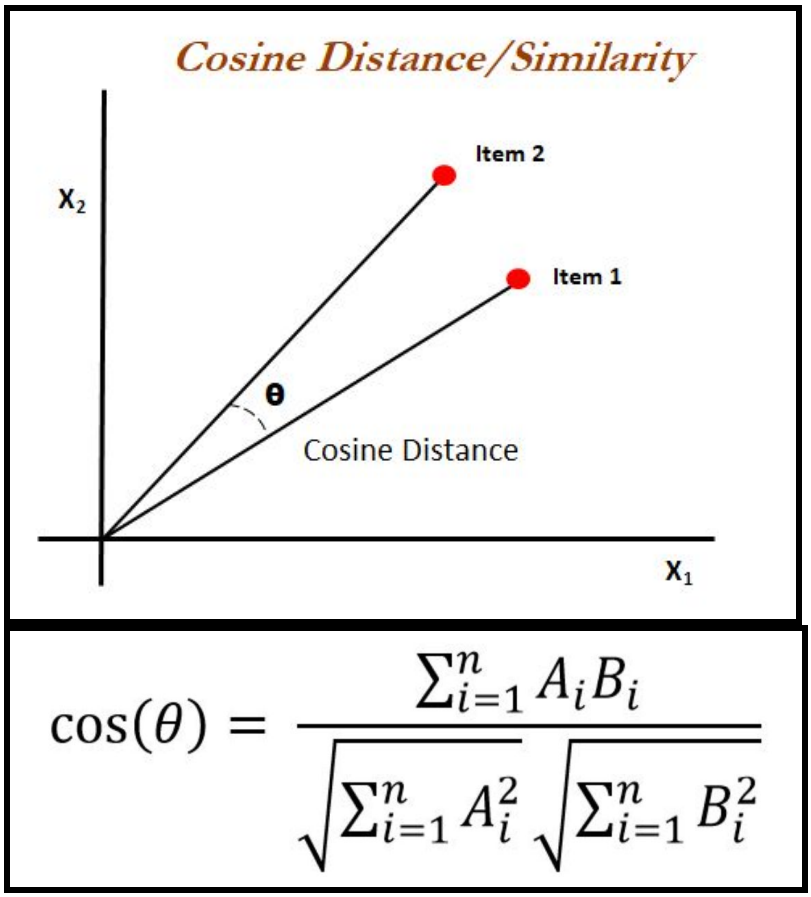
Now, we create different recommender systems and see how they differ from one another. One of the disadvantages of a content-based recommender system is that there are no concrete quantitative analysis metric we can use to measure its accuracy. Instead, I will use my familiarity with the videos to check the validity of these recommendations.

Based on Tags Used

First, we'll try to build a recommender system based on the tags used in each video. The more tags used in common, the more likely it is to be recommended to the user.

To do this, we create a TF-IDF model. TF-IDF stands for “Term-Frequency X Inverse Document Frequency”. TF-IDF is "essentially a measure of term importance, and how discriminative a word is in a corpus". Unlike a standard `CountVectorizer` model (another NLP technique), we don't just use the term frequency in a document of words in our vocabulary, we weigh its counts by 1 divided by its overall frequency. So that tags that show up often in ALL videos, are down weighted.

After building our TF-IDF model, we quantify the similarity between the vectors by calculating their cosine similarities.



Photos from "Statistics for Machine Learning" by Pratap Dangeti

As shown in the figure and equation above, cosine similarity is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in roughly the same direction.

Here are seven out of the top 20 recommendations given for one of the videos with the most views (and I am most familiar with regarding the artist and the genre), Canadian rapper Drake's song "God's Plan":

```
Recommendations for "Drake - God's Plan":
```

	title	channel_title	views	category	comment_count
162	Drake - God's Plan (Official Music Video)	OVO Sound	19586636	Music	66540
1660	Gods Plan - Drake (William Singe Cover)	William Singe	2004311	Music	3007
709	Guillermo – God's Plan	Jimmy Kimmel Live	4720264	Entertainment	3541
37	Drake - Nice For What	OVO Sound	60635812	Music	55653
6654	Drake's Dad Claps Back At Wendy Williams! He C...	Lailah Lynn	118828	Entertainment	741
3726	The epic late-night Fortnite stream featuring ...	ESPN	636473	Sports	1355
1263	DRAKE & NINJA PLAY DUOS ON FORTNITE! | Fortnit...	Twitch Moments	2663507	Gaming	1796

Right now, it looks pretty good. It includes covers of the songs, the artist's other songs, and other videos that involve Drake (whether it be gossip or activities involving the artist).

Recommending Videos from Different Categories

One of the disadvantages of a content-based recommender system is overspecialization--when a recommender system does not recommend items outside user's content profile. In this case, most of the videos recommended are most probably videos within the same category. To approach this, we build a recommender system that suggests similar videos (based on tag counts) that belong in different categories.

```
Recommendations for "The End of the F**king World | Official Trailer [HD] | Netflix":
```

	title	channel_title	views	category	comment_count
2027	THE CLOVERFIELD PARADOX | WATCH NOW | NETFLIX	Netflix	1575608	Entertainment	3032
459	The Kissing Booth | Official Trailer [HD] | Ne...	Netflix	7308023	Entertainment	5790
6508	Derren Brown: The Push I Official Trailer [HD]...	Netflix	135325	Entertainment	395
5128	Irreplaceable You | Official Trailer [HD] | Ne...	Netflix	316756	Entertainment	506
6991	The Moment George Clooney Met Amal | My Next G...	Netflix	91193	Entertainment	82

Overspecialization doesn't seem much of a problem for Drake's video since the recommender system recommended videos from other categories such as Entertainment, Sports and Gaming. But what about for the Netflix trailer for "The End of the F**king World"? The videos suggested all seem to belong to the Entertainment category.

```
Recommendations for "The End of the F**king World | Official Trailer [HD] | Netflix":
```

	title	views	category	comment_count
4072	Bright: What Went Wrong? – Wisecrack Edition	533207	Education	4618
5059	Reboot: The Guardian Code Official Trailer	326871	Film & Animation	5071
8085	'I have dad moves': Barack Obama discusses dan...	21700	News & Politics	70
2251	Ep4 It's on you and I | BTS: Burn the Stage	1378098	Music	1376
5194	HIS & HERS BOUJEE NIGHT OUT! VLOGMAS WEEK 1	304917	Howto & Style	1204

The recommender system successfully recommends videos from other categories. The first suggestion (therefore, most similar in tags), is the video "Bright: What Went Wrong? - Wisecrack Edition". This video falls under the Education category since it's a review of the Netflix show. By looking at the tags, it seems like the

main similarities between the two is the fact that they are both Netflix shows. This tells us that we must incorporate something other than just tags to calculate more accurate similarities.

Based on Description

Now, we check how well our recommender system does if we base it on the video's descriptions. For movie/show trailers, it might be more accurate since trailers usually include the actors, and the movie/show's plot in the description.

Recommendations for "Drake - God's Plan":

	title	channel_title	views	category	comment_count
156	Nicki Minaj - Barbie Tingz (Lyric Video)	NickiMinajAtVEVO	20262996	Music	40235
79	Nicki Minaj - Chun-Li	NickiMinajAtVEVO	36759844	Music	93136
1086	N.E.R.D, Rihanna - Lemon (Drake Remix - Audio)...	NERDVEVO	3126660	Music	2640
681	Nicki Minaj - Chun-Li (Live on SNL / 2018)	NickiMinajAtVEVO	4945185	Music	18683
13	Post Malone - Psycho ft. Ty Dolla $ign	PostMaloneVEVO	105629911	Music	45784

The new top recommended videos are different from the top recommended videos on the tag-based recommender system. All of the videos recommended now belong in the Music category. The recommended videos are now less about Drake and more on music videos that belong in the same genre as Drake's music.

Based on Video Description, Tags, and Video Channel

To capture the similarity of all these features, we can combine all of them and fit it into our TF-IDF model.

By doing so, we get these recommendations:

Recommendations for "Drake - God's Plan":

	title	channel_title	views	category	comment_count
1086	N.E.R.D, Rihanna - Lemon (Drake Remix - Audio)...	NERDVEVO	3126660	Music	2640
7329	Drake Bell - Rewind	DrakeBellVEVO	65973	Music	836
156	Nicki Minaj - Barbie Tingz (Lyric Video)	NickiMinajAtVEVO	20262996	Music	40235
37	Drake - Nice For What	OVO Sound	60635812	Music	55653
79	Nicki Minaj - Chun-Li	NickiMinajAtVEVO	36759844	Music	93136

The recommended videos are now videos that show more diversity in the topics and categories, while also still suggesting music from other artists who belong in the same genre.

Conclusion

In this project, I have built four content-based recommender systems based on different features. For the first two, we focus on the tags used in each video. For the second one, I only allow for recommendations outside the category. We see that this does not do very well because the videos don't seem to be very similar, except for the fact that they had very broad tags (such as "Netflix") in common. For the third one, we focus solely on the description. This can be more helpful for show or movie trailers since actors, directors and plots are often included in the description. Although, there is a higher chance of videos with little to no description will be paired with unrelated videos. So, for our fourth recommender system, we try to address all of these problems by combining the tags used, description, and channel title all in one string.

Limitations

The dataset used is very restricted; the videos in the dataset are only videos that were included in the daily "Top Trending List" in the United States and Great Britain for six months.