DS 5230 Unsupervised Learning

Final Project Report

Young Zhang, Xianrui She, Ka Chun Lee

August 19, 2021

Abstract

In this project, we hope to help create a better anime recommendation system for those streaming companies. To gain more insights into anime, we conducted a series of exploratory data analyses, including finding more popular anime genres, user rating distribution, and others. Later, we used K-means clustering, Popularity based filtering, Content-based filtering, and Collaborative filtering(SVD) to predict user preference and recommend anime, achieving significant test set performance, while validating recommendations given by different methods.

Introduction

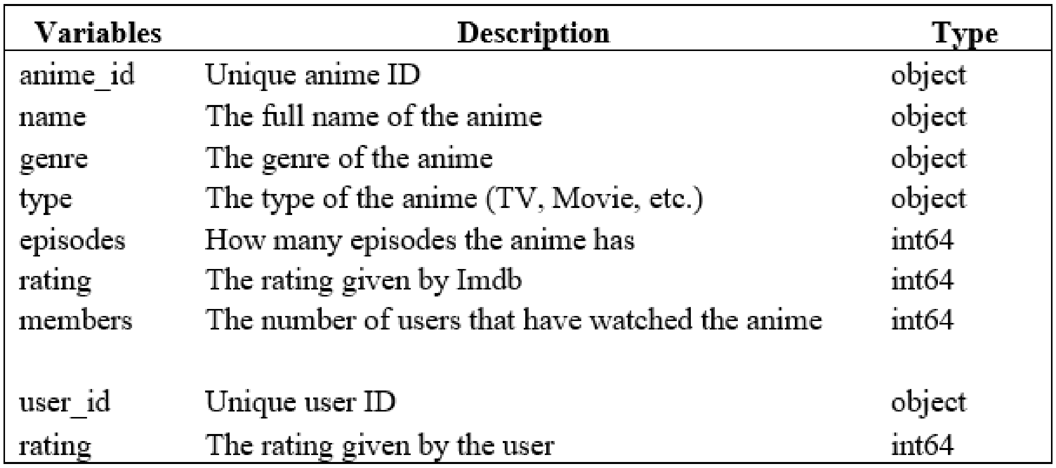
In its most basic form, anime, specifically Japanese anime, is a type of animation that can be hand-drawn or created with computers. It is very similar to American cartoons but uses signature aspects of Japanese-style animation that's very different from the western-styled cartoon including vibrant colors, dramatic panning, and characteristic facial expressions which makes anime unique (Bond 2018). For decades, Japanese anime was only the domain of small minorities. Thanks to economic globalization, many excellent anime series have reached American shores and became prominent on many popular streaming sites like Netflix, Hulu, etc.

Our targeted clients are these streaming companies that need improvements or suggestions on their recommender system to help create a better streaming platform and browsing experiences for the users and boost playing time and revenue for the streaming companies. We also aim to provide better insights for the anime production studios to understand the community better.

Data Description

The dataset used in this project was obtained from Kaggle:
https://www.kaggle.com/CooperUnion/anime-recommendations-database

There is an anime table that contains 12,294 records and 7 features and an user rating table that contains 7,813,737 records and 3 features. The anime table contains all the anime names with their own features and the user rating table has all the users' rating information. A detailed description of the features is listed in the table below.



Variables	Description	Type
anime_id	Unique anime ID	object
name	The full name of the anime	object
genre	The genre of the anime	object
type	The type of the anime (TV, Movie, etc.)	object
episodes	How many episodes the anime has	int64
rating	The rating given by Imdb	int64
members	The number of users that have watched the anime	int64
user_id	Unique user ID	object
rating	The rating given by the user	int64

Methodology

Python was the programming language used for tidying the data, data visualization, and model analysis in this project. We imported NumPy and Pandas libraries to help with the data cleaning and preprocessing. Matplotlib and seaborn libraries were used for EDA and data visualization, etc. Lastly, we used the scikit-learn package for model analysis. Since we also aimed to produce a recommender system, clustering was first used based on the dataset's characteristics. Then we examined popularity-based filtering, content-based filtering, and collaborative filtering for different recommendation needs.

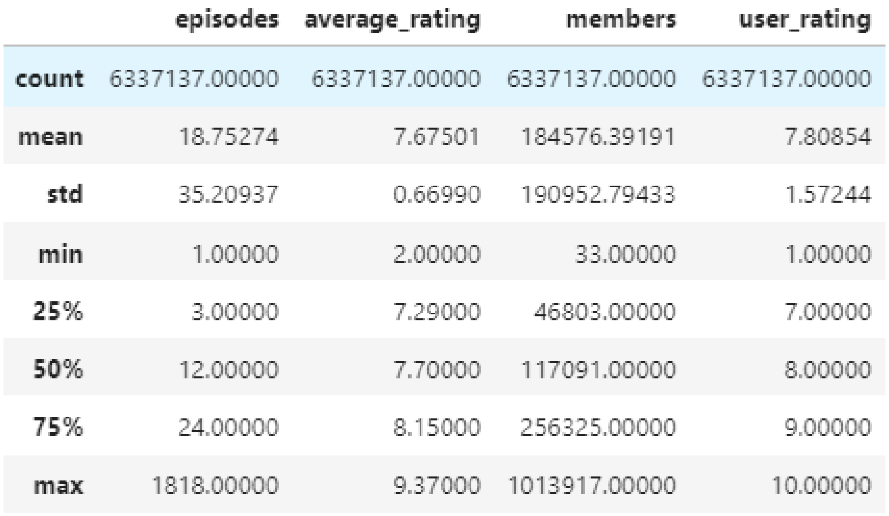
Data Cleaning

For a more straightforward data processing, we decided to clean the two tables separately and merge them afterward for future applications. Looking at the anime table, the first step was text cleaning using regular expression operations because the anime name contains many special characters and symbols. Next, we dealt with the missing values. Since the attributes for each anime are unique, and there are not that many values missing, we decided to remove them altogether. Then we classified the datatype mainly for the exploratory data analysis and modeling later on. Lastly, we reset the indexes, so that the index matches the row count. Similar processes were also performed to the user rating table with only one exception, the rating column. We observed many '-1' ratings, meaning that the user has watched the anime but left no ratings. It is unreasonable to remove them as they affect the total playing time. So, we kept them while ignoring them when calculating the average rating and when performing the rating-based applications.

After the data cleaning process, we merged the two tables and checked any duplicated user-anime pairs. After removing all 7 of them, the dimension of the final dataset contains 7,813,600 records and 9 features with no duplicated and missing data.

Exploratory Data Analysis

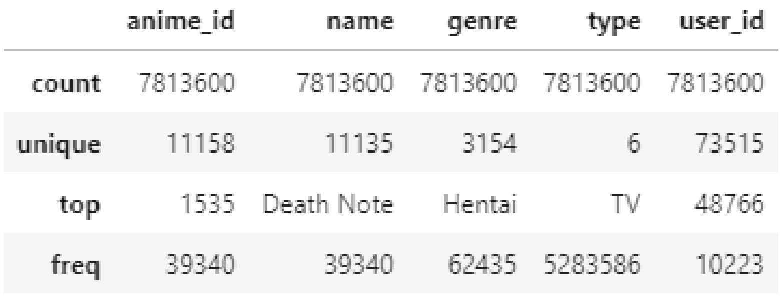
A concise summary of the numerical data is shown in **Table 1**. It can be inferred that the minimum average rating is 2 while the maximum is 9.37. The community size of the anime ranges from 33 to a little over one million. The number of episodes ranges from 1 to 1,818, and the user rating ranges from 1 to 10, with an average value of 7.81.



	episodes	average_rating	members	user_rating
count	6337137.00000	6337137.00000	6337137.00000	6337137.00000
mean	18.75274	7.67501	184576.39191	7.80854
std	35.20937	0.66990	190952.79433	1.57244
min	1.00000	2.00000	33.00000	1.00000
25%	3.00000	7.29000	46803.00000	7.00000
50%	12.00000	7.70000	117091.00000	8.00000
75%	24.00000	8.15000	256325.00000	9.00000
max	1818.00000	9.37000	1013917.00000	10.00000

Table 1 - Concise summary of the numerical data

For the categorical data shown in **Table 2**, we find there are 11,158 unique anime id with Death Note being the most popular anime. It was watched by a total of 39,340 users. There are 3,154 unique genre combinations with Hentai being the most frequent genre. TV is the most repeated anime type with a frequency of over 5.2 million. Lastly, there are 73,515 unique users and the user with id 48766 have watched over 10,000 anime.



	anime_id	name	genre	type	user_id
count	7813600	7813600	7813600	7813600	7813600
unique	11158	11135	3154	6	73515
top	1535	Death Note	Hentai	TV	48766
freq	39340	39340	62435	5283586	10223

Table 2 - Concise summary of the categorical data

A heatmap shown in **Figure 1** was also generated to identify any correlations between the numerical variables. There is a positive correlation with community size and average rating. It is expected because as the community size increases, i.e., an anime becomes more popular, the average rating is also likely to increase.

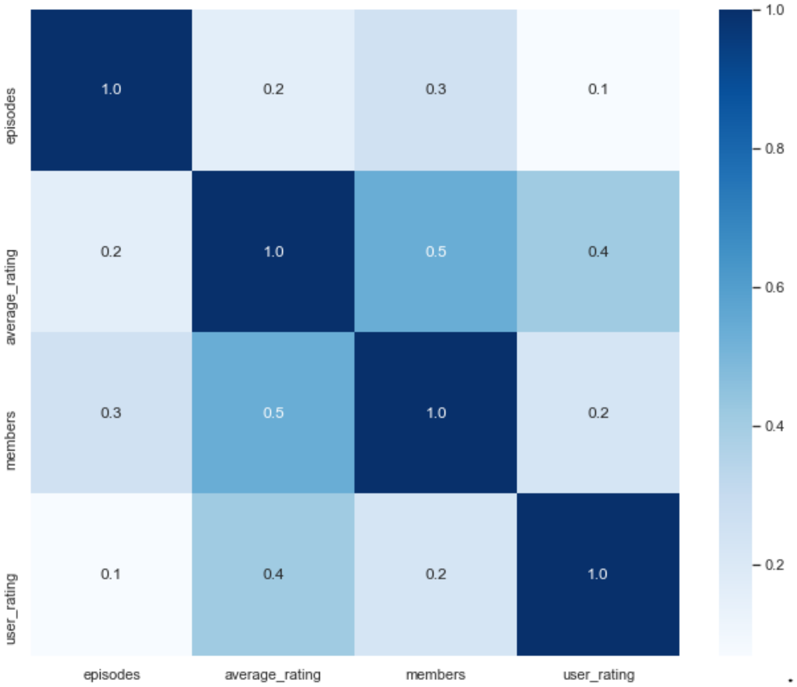
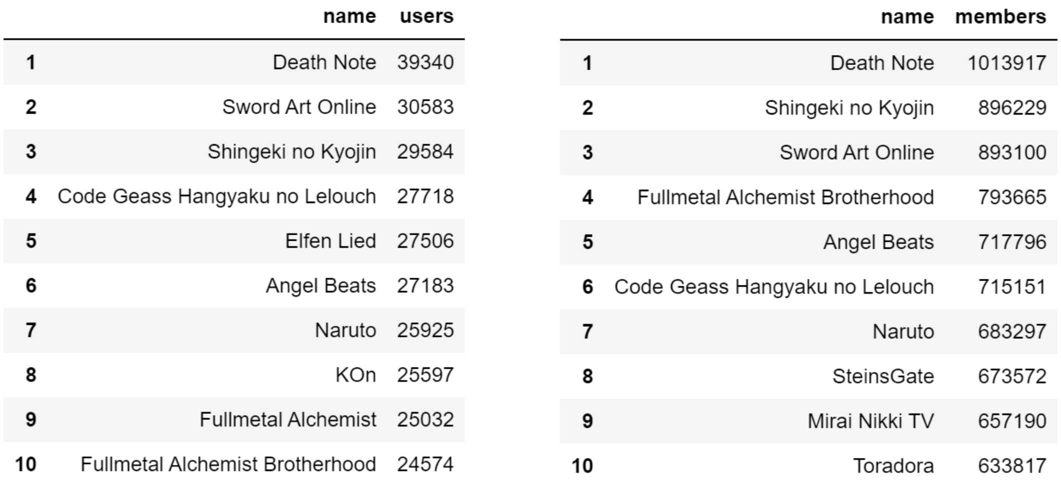


Figure 1 - Heat map of the anime dataset

Next, we are interested in finding out the most popular anime by different metrics. Specifically, we listed the top ten anime by the number of users, the number of members in the respective communities, and by the average rating given by our dataset. The results are given by the tables below.

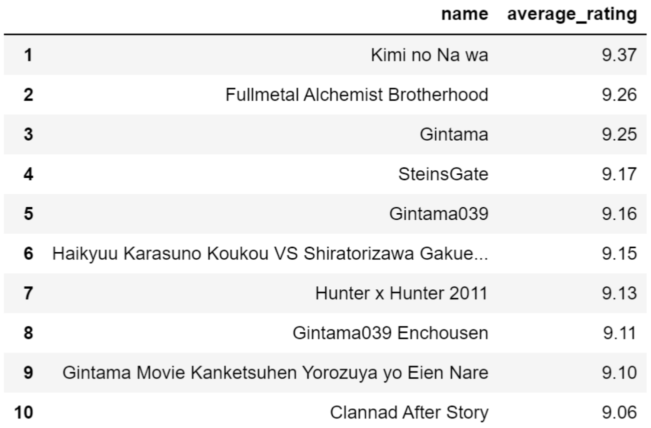


	name	users
1	Death Note	39340
2	Sword Art Online	30583
3	Shingeki no Kyojin	29584
4	Code Geass Hangyaku no Lelouch	27718
5	Elfen Lied	27506
6	Angel Beats	27183
7	Naruto	25925
8	KOn	25597
9	Fullmetal Alchemist	25032
10	Fullmetal Alchemist Brotherhood	24574

	name	members
1	Death Note	1013917
2	Shingeki no Kyojin	896229
3	Sword Art Online	893100
4	Fullmetal Alchemist Brotherhood	793665
5	Angel Beats	717796
6	Code Geass Hangyaku no Lelouch	715151
7	Naruto	683297
8	SteinsGate	673572
9	Mirai Nikki TV	657190
10	Toradora	633817

Table 3 - Top 10 popular anime by number of users and members

The two tables above in **Table 3**, which use the number of users and the size of the community, showed similar results. For example, “Death Note” came on top on both lists, while several others appeared in both top ten lists as well. On the other hand, the result from average rating ended up different from the other two.



	name	average_rating
1	Kimi no Na wa	9.37
2	Fullmetal Alchemist Brotherhood	9.26
3	Gintama	9.25
4	SteinsGate	9.17
5	Gintama039	9.16
6	Haikyuu Karasuno Koukou VS Shiratorizawa Gakue...	9.15
7	Hunter x Hunter 2011	9.13
8	Gintama039 Enchousen	9.11
9	Gintama Movie Kanketsuhen Yorozuya yo Eien Nare	9.10
10	Clannad After Story	9.06

Table 4 - Top 10 popular anime by average rating

As demonstrated in **Table 4**, the top ten anime are completely changed when using average ratings to rank. This is to be expected because this list takes consideration of the quality of an anime. If an anime is made of high quality but missed the audience for any reason, it may receive a better rating than that of worse quality but watched by many because of its fame.

Community size is an important part of our dataset, so we decided to dig into it furthermore. First, we examined the distribution of anime community sizes by constructing a histogram like below

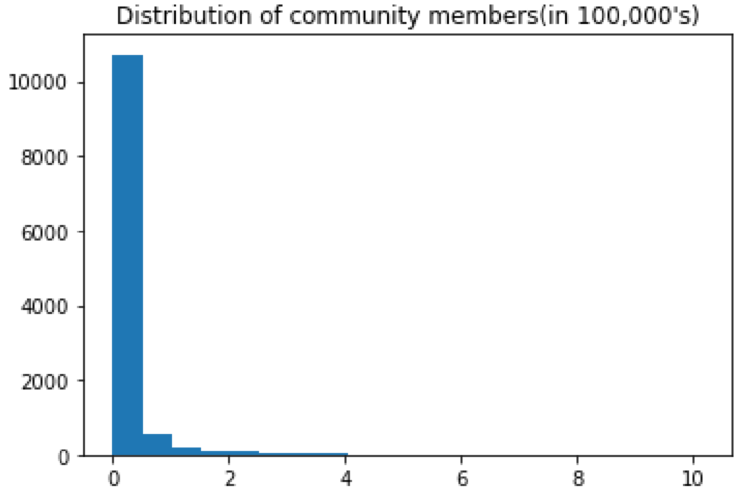


Figure 2 - Distribution of community members(in 100,000's)

Based on the histogram in **Figure 2**, we can clearly see that most anime has low numbers of members while very few have sizes greater than 100,000 members. This finding is further reinforced by **Figure 3.**

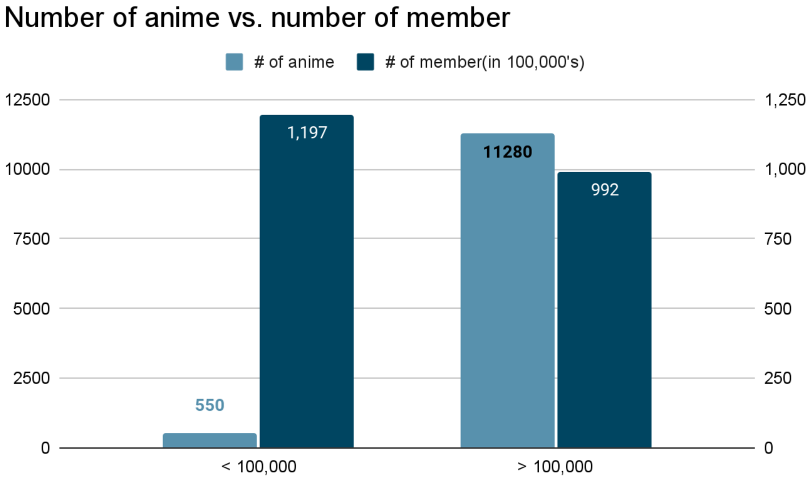


Figure 3 - Contribution of community members

550 of all the anime contributed 1,197 hundred thousand members, which is more than half of the total number, while 11,280 anime accounted for the rest.

Genre breakdown

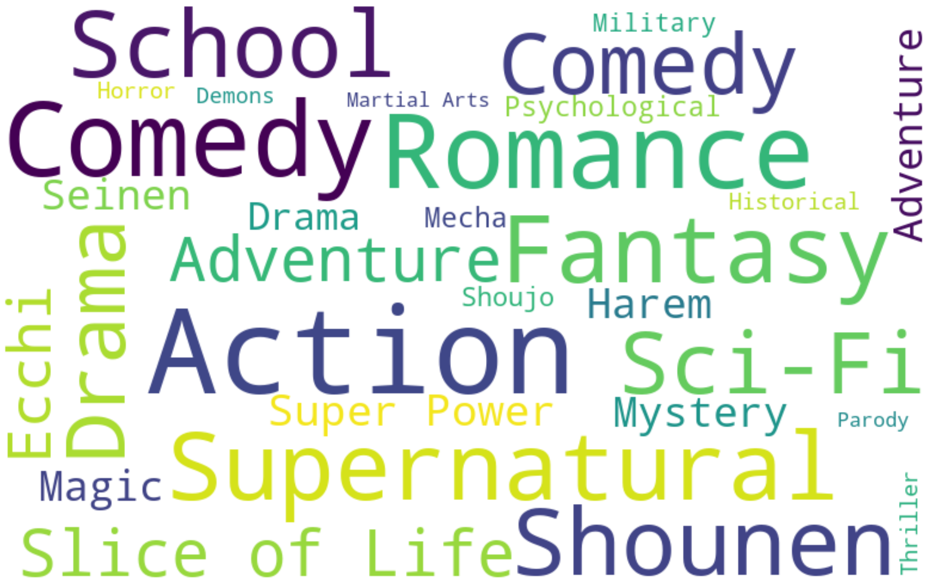


Figure 4 - Word cloud on genre

Next, we turned our attention to the different genres by building a word cloud on it. The metric used is the number of users for anime in each genre. According to **Figure 4**, there are clearly more popular genres compared to the other. Specifically, *comedy, romance, action, fantasy*, and *school* ranked on top, whereas *horrors, demons, parody, historical*, and *martial art* are watched the least. In general, users favor genres that are lighter and more entertaining over ones that are more serious and horrifying.

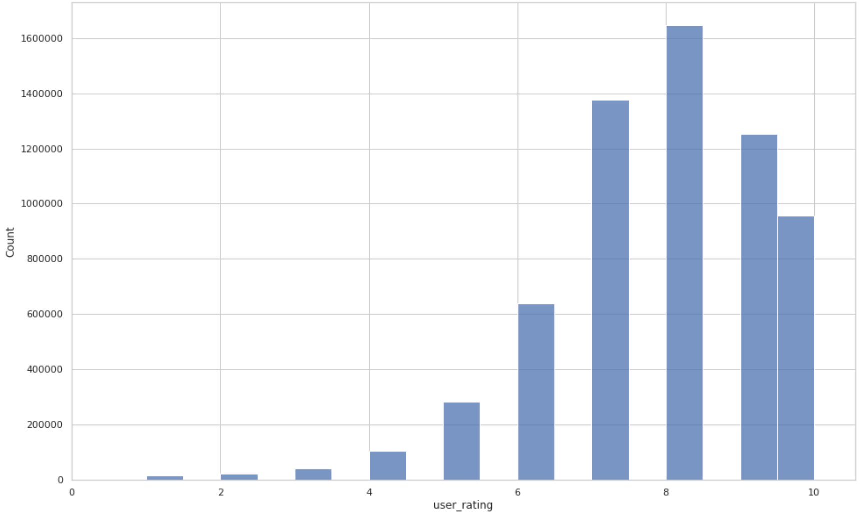


Figure 5 - Distribution of user-rating

In **Figure 5**, user-rating is a heavily left-skewed distribution, and 8 points rating is the most frequent since there are roughly 1.6 million users who give 8/10 rating . The following is the 7/10 rating which is roughly

1.4 million. The least frequent is 1/10 rating and there is only roughly 20000.We can observe that users are normally satisfied with anime since people are inclined to rate anime between 7 to 9 rating.

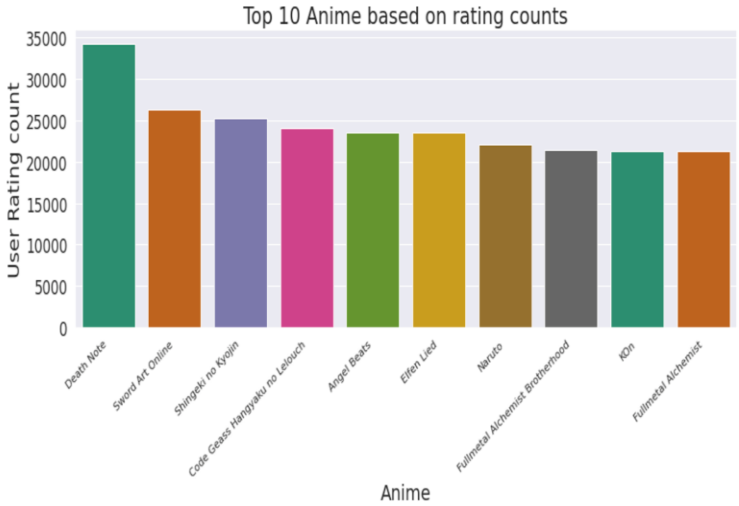


Figure 6 - Distribution of user-rating counts and Anime

It can be observed from **Figure 6** that the Death Note is the most popular Anime which has roughly 34,000 user rating counts followed by the Sword art online. The result is very consistent with the top ten anime tables shown in **Table 3**.

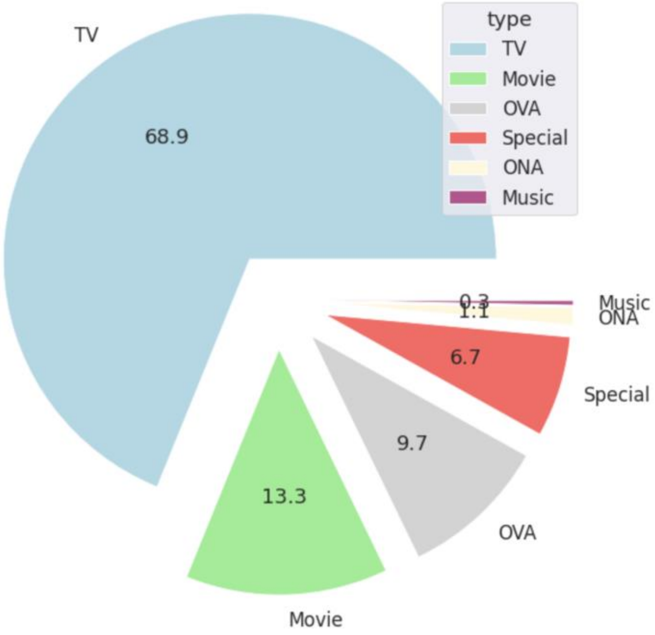


Figure 7 - type pie chart

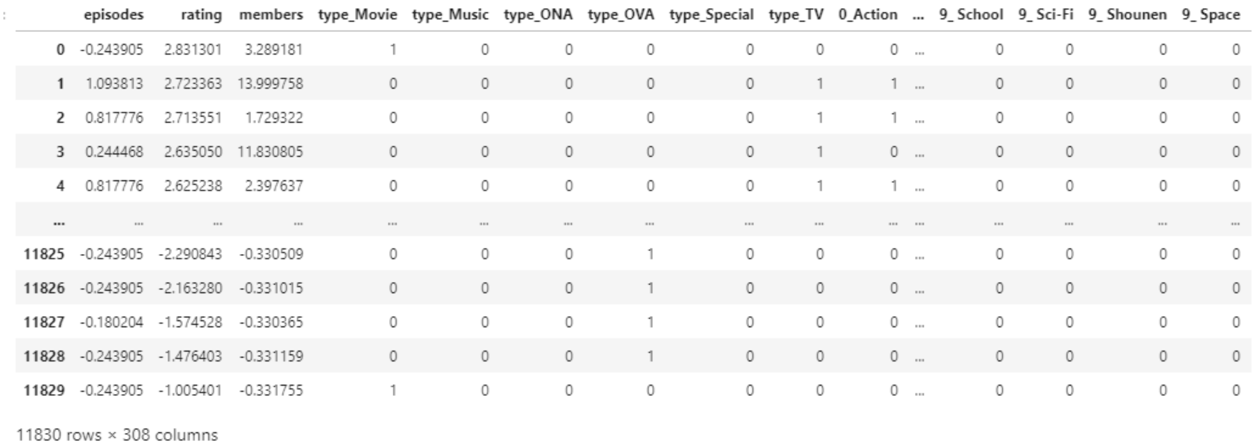
Figure 7 presents 6 unique anime types including TV, Movie, OVA, Special, ONA, and Music. Among them, 68.9% of the anime was shown on TV, 13.3% are movies, 9.7% are OVA, 6.7% are labeled specials, 1.1% are ONA and 0.2% are Music contents.

Clustering

K-mean clustering is one of the most simple and powerful unsupervised machine learning techniques that divides a dataset into groups with similar characteristics. In this project, clustering was first used to group similar anime together. The cluster-based recommendation can be helpful when recommending to new users with no watch history.

-*Data Preprocessing*

The genre column of the dataset was first expanded because each anime could have more than one genre, which are collapsed into a single cell. Then one hot encoding was used for the categorical variables such as the genre and type columns. Lastly, numerical variables such as episodes, ratings, and member count were scaled using the standard scaling method. The final dataset is shown in **Table 5**.



	episodes	rating	members	type_Movie	type_Music	type_ONA	type_OVA	type_Special	type_TV	0_Action	...	9_ School	9_ Sci-Fi	9_ Shounen	9_ Space
0	-0.243905	2.831301	3.289181	1	0	0	0	0	0	0	...	0	0	0	0
1	1.093813	2.723363	13.999758	0	0	0	0	0	1	1	...	0	0	0	0
2	0.817776	2.713551	1.729322	0	0	0	0	0	1	1	...	0	0	0	0
3	0.244468	2.635050	11.830805	0	0	0	0	0	1	0	...	0	0	0	0
4	0.817776	2.625238	2.397637	0	0	0	0	0	1	1	...	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
11825	-0.243905	-2.290843	-0.330509	0	0	0	1	0	0	0	...	0	0	0	0
11826	-0.243905	-2.163280	-0.331015	0	0	0	1	0	0	0	...	0	0	0	0
11827	-0.180204	-1.574528	-0.330365	0	0	0	1	0	0	0	...	0	0	0	0
11828	-0.243905	-1.476403	-0.331159	0	0	0	1	0	0	0	...	0	0	0	0
11829	-0.243905	-1.005401	-0.331755	1	0	0	0	0	0	0	...	0	0	0	0

11830 rows × 308 columns

Table 5 - Data used for clustering after preprocessing steps

For determining the optimal number of clusters, the silhouette method was used. A range of candidate values of k (number of clusters) was picked and trained K-Means clustering for each of the values of k. For each k-Means clustering model, the silhouette scores were plotted and observed in **Figure 8**. The graph has an obvious elbow point when k=4, which will be the optimal cluster size. The resulting clusters were cast on a 2-d grid shown in **Figure 9**. The top 5 anime ranked by average rating in each cluster is shown in **Figure 10**. We find k-mean clustering a good choice for recommendations to users who don't have any watch history. The recommendation is solely based on which cluster the anime belongs to, and the suggestion would be the top 5 anime within that same cluster.

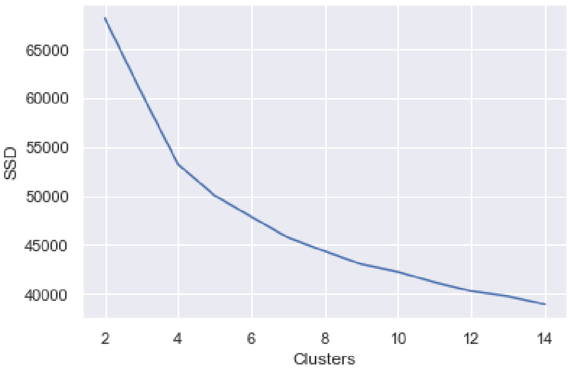


Figure 8 - Silhouette score vs number of clusters

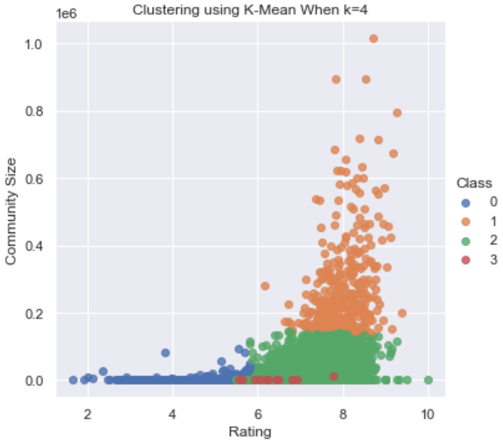
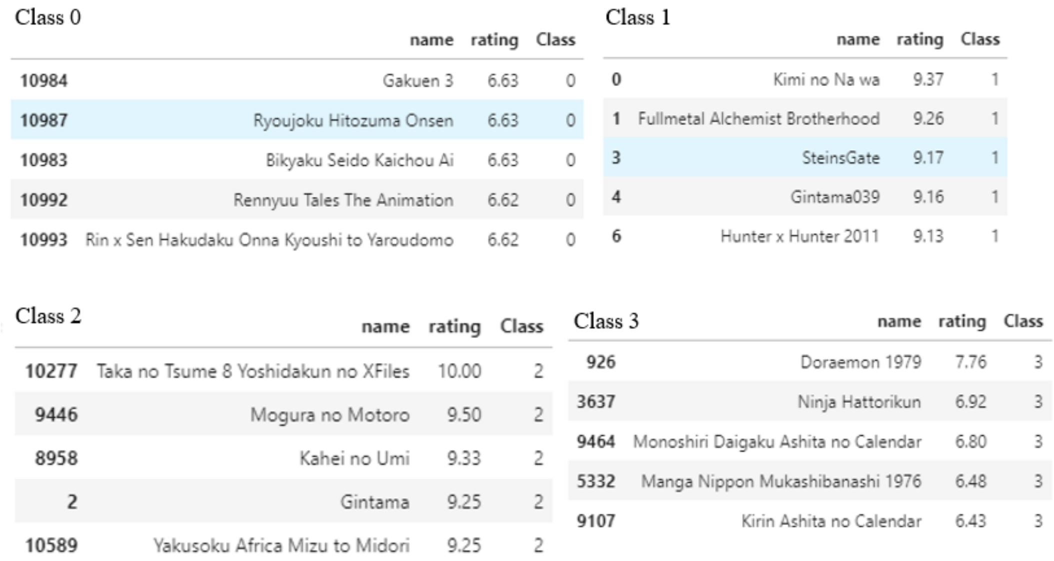


Figure 9 - Cluster visualization


Class 0

	name	rating	Class
10984	Gakuen 3	6.63	0
10987	Ryoujoku Hitozuma Onsen	6.63	0
10983	Bikyaku Seido Kaichou Ai	6.63	0
10992	Rennyuu Tales The Animation	6.62	0
10993	Rin x Sen Hakudaku Onna Kyoushi to Yaroudomo	6.62	0

Class 1

	name	rating	Class
0	Kimi no Na wa	9.37	1
1	Fullmetal Alchemist Brotherhood	9.26	1
3	SteinsGate	9.17	1
4	Gintama039	9.16	1
6	Hunter x Hunter 2011	9.13	1

Class 2

	name	rating	Class
10277	Taka no Tsume 8 Yoshidakun no XFiles	10.00	2
9446	Mogura no Motoro	9.50	2
8958	Kahei no Umi	9.33	2
2	Gintama	9.25	2
10589	Yakusoku Africa Mizu to Midori	9.25	2

Class 3

	name	rating	Class
926	Doraemon 1979	7.76	3
3637	Ninja Hattorikun	6.92	3
9464	Monoshiri Daigaku Ashita no Calendar	6.80	3
5332	Manga Nippon Mukashibanashi 1976	6.48	3
9107	Kirin Ashita no Calendar	6.43	3

Figure 10 - Top 5 anime ranked by average rating in each cluster

Recommender System

The purpose of the recommendation system is to predict the 'rating' and 'preference' that a user would give to an anime. We used three different methods: Popularity-based filtering, Content-based filtering, and Collaborative filtering to filter out all anime for users and rank them to serve different types of users.

-Popularity based filtering

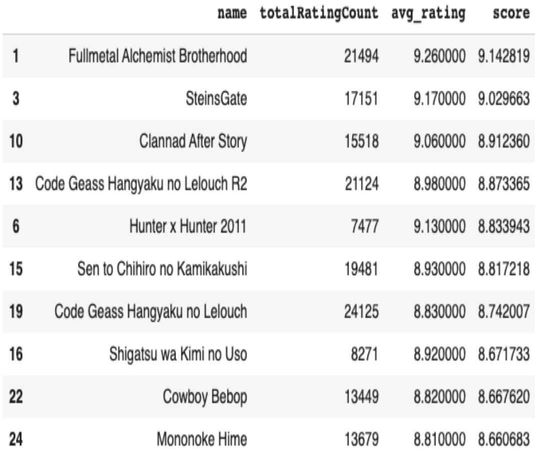
The purpose of the popularity-based filtering is to avoid purely using rating as the metric since we don't want an anime with a rating of 9 from only 10 voters to be considered 'better' than an anime with a rating of 8.9 from 10,000 voters. Purely using rating as a metric could lead to bias because the number of ratings matters. Therefore, we used the weighted rating to ensure an anime with a 9 rating from 500000 voters weighs more than an anime with the same rating but much fewer vote counts.

$$WeightedRating(\mathbf{WR}) = \left(\frac{\mathbf{v}}{\mathbf{v}+\mathbf{m}} \cdot \mathbf{R}\right) + \left(\frac{\mathbf{m}}{\mathbf{v}+\mathbf{m}} \cdot \mathbf{C}\right)$$

Equation 1 - Weighted Rating

Popularity based filtering is based on the Equation 1 Weighted Rating function.
In the above equation,
V:the number of votes for the anime (‘total rating count’ in our dataset)
R:the average rating of the anime(‘average rating’ in our dataset)
m:is the minimum votes required to be listed in the chart
C:the mean vote across the whole report. (Mean of the sum of ‘average rating’)

‘m’ is the hyperparameter that we could choose and control. In our project, we choose to use 85% quantile to be the ‘m’. In other words, we would only consider the top 15% of the anime in terms of the number of votes garnered. Based on the operation, we will define a new feature ‘score’ and add it into the dataframe. Then, we sort the dataframe in descending order through score columns. The following is the top 10 anime based on weighted rating function. The result shows that Fullmetal Alchemist Brotherhood has the highest score, followed by SteinsGate, and Clannad After Story.



	name	totalRatingCount	avg_rating	score
1	Fullmetal Alchemist Brotherhood	21494	9.260000	9.142819
3	SteinsGate	17151	9.170000	9.029663
10	Clannad After Story	15518	9.060000	8.912360
13	Code Geass Hangyaku no Lelouch R2	21124	8.980000	8.873365
6	Hunter x Hunter 2011	7477	9.130000	8.833943
15	Sen to Chihiro no Kamikakushi	19481	8.930000	8.817218
19	Code Geass Hangyaku no Lelouch	24125	8.830000	8.742007
16	Shigatsu wa Kimi no Uso	8271	8.920000	8.671733
22	Cowboy Bebop	13449	8.820000	8.667620
24	Mononoke Hime	13679	8.810000	8.660683

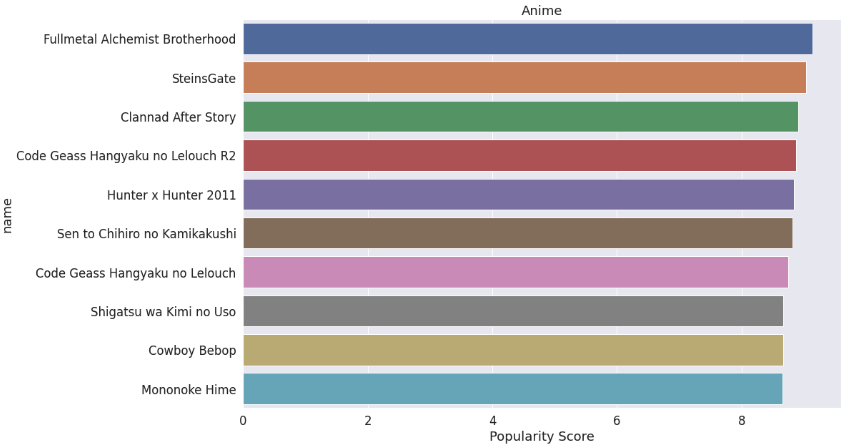
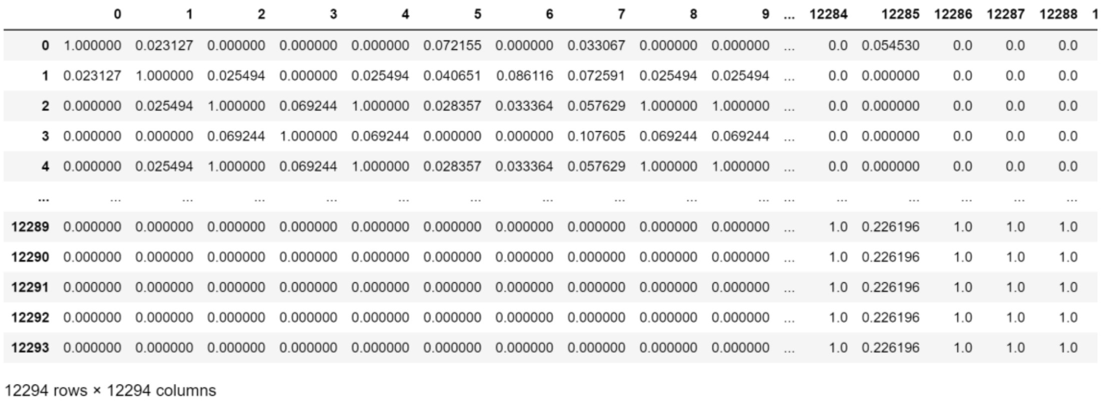


Table 6 - Top 10 anime ranked by weighted rating

-Content-based filtering

Next, we conducted content-based filtering based on genre similarity. Compared to other filtering techniques, content-based only uses item similarities, recommending users similar items to what they already chose. In our project, we employed the basic concept of content-based filtering using only genre as the item feature.

Before implementation, we first processed the anime-genre relationship into a TF-IDF matrix. Doing so allows us to better account for words that appear less often but poseese high significance. Next, we constructed a square matrix representing pair-wise item similarity using cosine similarity. The matrix is represented as **Table 7** below

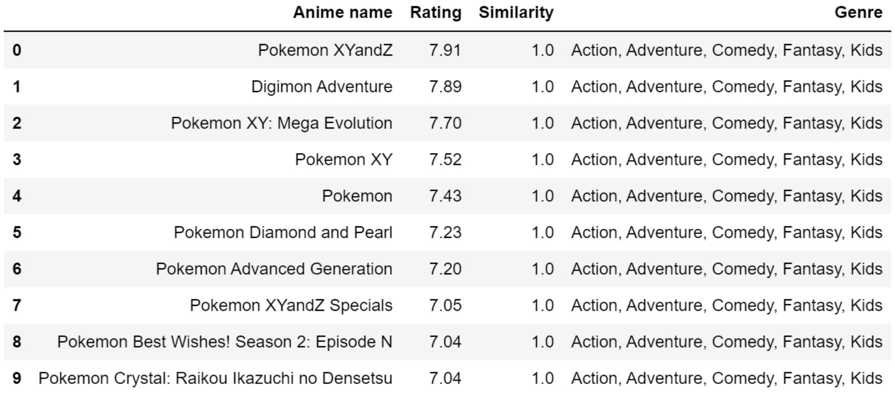


	0	1	2	3	4	5	6	7	8	9	...	12284	12285	12286	12287	12288	1
0	1.000000	0.023127	0.000000	0.000000	0.000000	0.072155	0.000000	0.033067	0.000000	0.000000	...	0.0	0.054530	0.0	0.0	0.0	
1	0.023127	1.000000	0.025494	0.000000	0.025494	0.040651	0.086116	0.072591	0.025494	0.025494	...	0.0	0.000000	0.0	0.0	0.0	
2	0.000000	0.025494	1.000000	0.069244	1.000000	0.028357	0.033364	0.057629	1.000000	1.000000	...	0.0	0.000000	0.0	0.0	0.0	
3	0.000000	0.000000	0.069244	1.000000	0.069244	0.000000	0.000000	0.107605	0.069244	0.069244	...	0.0	0.000000	0.0	0.0	0.0	
4	0.000000	0.025494	1.000000	0.069244	1.000000	0.028357	0.033364	0.057629	1.000000	1.000000	...	0.0	0.000000	0.0	0.0	0.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
12289	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	1.0	0.226196	1.0	1.0	1.0	
12290	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	1.0	0.226196	1.0	1.0	1.0	
12291	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	1.0	0.226196	1.0	1.0	1.0	
12292	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	1.0	0.226196	1.0	1.0	1.0	
12293	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	1.0	0.226196	1.0	1.0	1.0	

12294 rows × 12294 columns

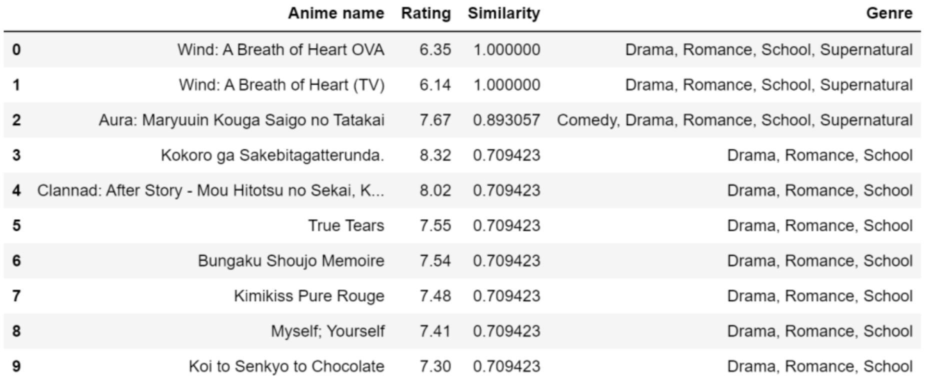
Table 7 - Pair-wise item similarity

Using the values in **Table 8**, we are able to come up with the top ten similar anime with a single input. For example, when the user watches “Pokemon” and “Kimi no Nawa”, the recommendations are as follows



	Anime name	Rating	Similarity	Genre
0	Pokemon XYandZ	7.91	1.0	Action, Adventure, Comedy, Fantasy, Kids
1	Digimon Adventure	7.89	1.0	Action, Adventure, Comedy, Fantasy, Kids
2	Pokemon XY: Mega Evolution	7.70	1.0	Action, Adventure, Comedy, Fantasy, Kids
3	Pokemon XY	7.52	1.0	Action, Adventure, Comedy, Fantasy, Kids
4	Pokemon	7.43	1.0	Action, Adventure, Comedy, Fantasy, Kids
5	Pokemon Diamond and Pearl	7.23	1.0	Action, Adventure, Comedy, Fantasy, Kids
6	Pokemon Advanced Generation	7.20	1.0	Action, Adventure, Comedy, Fantasy, Kids
7	Pokemon XYandZ Specials	7.05	1.0	Action, Adventure, Comedy, Fantasy, Kids
8	Pokemon Best Wishes! Season 2: Episode N	7.04	1.0	Action, Adventure, Comedy, Fantasy, Kids
9	Pokemon Crystal: Raikou Ikazuchi no Densetsu	7.04	1.0	Action, Adventure, Comedy, Fantasy, Kids

Table 8 - Top 10 recommendations for “Pokemon”
users using content-based filtering



	Anime name	Rating	Similarity	Genre
0	Wind: A Breath of Heart OVA	6.35	1.000000	Drama, Romance, School, Supernatural
1	Wind: A Breath of Heart (TV)	6.14	1.000000	Drama, Romance, School, Supernatural
2	Aura: Maryuuin Kouga Saigo no Tatakai	7.67	0.893057	Comedy, Drama, Romance, School, Supernatural
3	Kokoro ga Sakebitagatterunda.	8.32	0.709423	Drama, Romance, School
4	Clannad: After Story - Mou Hitotsu no Sekai, K...	8.02	0.709423	Drama, Romance, School
5	True Tears	7.55	0.709423	Drama, Romance, School
6	Bungaku Shoujo Memoire	7.54	0.709423	Drama, Romance, School
7	Kimikiss Pure Rouge	7.48	0.709423	Drama, Romance, School
8	Myself; Yourself	7.41	0.709423	Drama, Romance, School
9	Koi to Senkyo to Chocolate	7.30	0.709423	Drama, Romance, School

Table 9 - Top 10 recommendations for “Kimi no Nawa”

users using content-based filtering

-*Collaborative filtering*

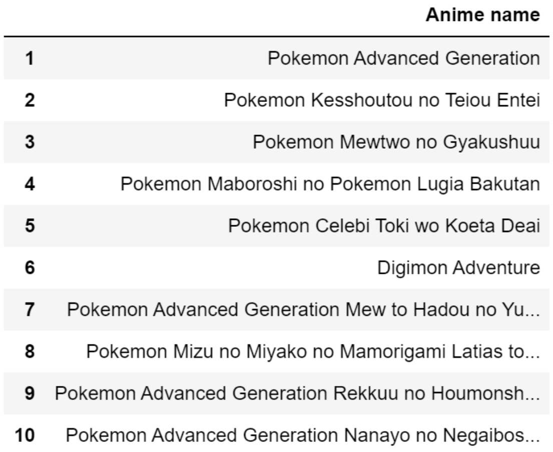
Compared to content-based filtering, collaborative filtering takes consideration of the entire collection of users and their preferences so that it gives recommendations based on user similarities. There are two main goals with collaborative filtering in our recommendation system. One is to provide anime recommendations to users with given anime input, similarly to content-based filtering. The other is to provide user-anime specific rating prediction, which allows us to forecast the rating a user would give to an anime he/she has not watched yet.

To start off, we first filtered out anime that have less than 500 users so that insignificant ones would not influence the model. We feel that 500 is an appropriate cut-off point given the average number of users for an anime is way higher.

Then, using python library *surprise*, we constructed the anime-user matrix with individual ratings in each cell, using which for matrix factorization. Doing so allows us to employ different machine learning models to build the collaborative-based recommendation system.

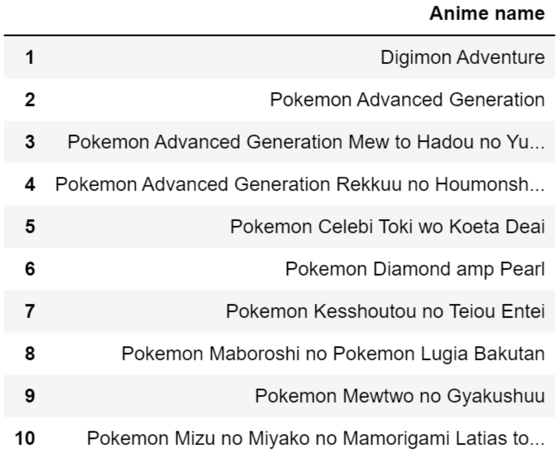
With provided model selections, we chose several to implement. In particular, we implemented KNN and SVD for recommending anime while using two additional models for user rating prediction.

With KNN and SVD recommenders, we tested with the same input as the content-based filtering, which is “Pokemon”. The results are shown in **Table 10** and **Table 11** below



	Anime name
1	Pokemon Advanced Generation
2	Pokemon Kesshoutou no Teiou Entei
3	Pokemon Mewtwo no Gyakushuu
4	Pokemon Maboroshi no Pokemon Lugia Bakutan
5	Pokemon Celebi Toki wo Koeta Deai
6	Digimon Adventure
7	Pokemon Advanced Generation Mew to Hadou no Yu...
8	Pokemon Mizu no Miyako no Mamorigami Latias to...
9	Pokemon Advanced Generation Rekkuu no Houmonsh...
10	Pokemon Advanced Generation Nanayo no Negaibos...

Table 10 - Top 10 recommendations for “Pokemon” using KNN



	Anime name
1	Digimon Adventure
2	Pokemon Advanced Generation
3	Pokemon Advanced Generation Mew to Hadou no Yu...
4	Pokemon Advanced Generation Rekkuu no Houmonsh...
5	Pokemon Celebi Toki wo Koeta Deai
6	Pokemon Diamond amp Pearl
7	Pokemon Kesshoutou no Teiou Entei
8	Pokemon Maboroshi no Pokemon Lugia Bakutan
9	Pokemon Mewtwo no Gyakushuu
10	Pokemon Mizu no Miyako no Mamorigami Latias to...

Table 11 - Top 10 recommendations for “Pokemon” using SVD

The top 10 recommendations are very similar to that of the content-based filtering. This would mean that users do like watching similar entries in the same anime series together, as demonstrated by the serial anime “Pokemon”.

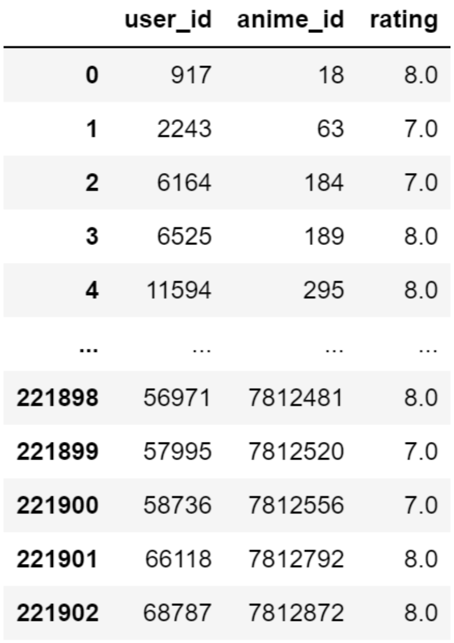
For user rating prediction, we considered KNN, SVD, SlopeOne, and Non-negative Matrix Factorization(NMF) models. To choose the best one among the four models, we cross validated all four models on the dataset and computed average test set RMSEs as the measuring metric. The results are displayed in the table below

	Test RMSE	Train time	Test time
SVD	1.930	46.173	2.061
KNNWithZScore	2.005	21.771	98.629
SlopeOne	2.022	23.476	103.182
NMF	2.165	54.137	2.255

Table 12 - Cross validation results

Among the four models, SVD showed the best RMSE performance as well as on-average good run-time, thus we decided to use SVD as our model to predict user ratings.

After splitting our dataset into training and testing, we re-run the SVD model to obtain our final test set RMSE, equaling 1.095. In comparison, the baseline RMSE, which is calculated by using the difference between each individual rating and the average rating from the column average_rating in the dataset, turned out to be 3.951. With the test set RMSE using SVD being smaller than the baseline value, we can conclude that our model is effective. Below is a glimpse at the user rating predictions made by the model



	user_id	anime_id	rating
0	917	18	8.0
1	2243	63	7.0
2	6164	184	7.0
3	6525	189	8.0
4	11594	295	8.0
...	...	...	...
221898	56971	7812481	8.0
221899	57995	7812520	7.0
221900	58736	7812556	7.0
221901	66118	7812792	8.0
221902	68787	7812872	8.0

Table 13 -Sample user rating prediction

Result/Conclusion

In conclusion, our project aims to recommend different anime to new-user and current users, based on our methods of filtering. The K-mean clustering and Popularity based filtering are to handle new-users, especially for those users without watch preferences and history. If they don't have any searching and watching history, the recommendation system will recommend the hottest and highest rating anime through K-mean clustering and Popularity based filtering method.

Additionally, we would recommend animes based on content-based filtering if we have their searching history or watching history. The recommendation system will provide similar anime by item similarities for those current users. For example, if a person watched “Pokemon” before, the recommendation system will provide anime similar to Pokemon. Apart from using content-based filtering, collaborative filtering takes consideration of the entire collection of users and their preferences so that it gives recommendations based on user similarities.

Overall, our project can serve most of the users, but there is a limitation to it. Since our dataset does not have published time of different anime, it may cause a problem that recommendation systems may recommend some outdated anime but with high ratings to users. If there is further data which provides publish time for every anime, we can offer some updated anime to users , better predict user-preference, and provide a better service to retain current users and attract new users.

Furthermore, we could improve our models in complexity. For example, our content-based filter only used genre as independent variables, due to how many different genre labels there are. To improve on this, we could incorporate all the variables included in the dataset and perform dimension reduction to get even more accurate item similarities. For clustering, we could also implement more clustering methods other than K-means, such as DBScan, hierarchical clustering, and other methods studied in lectures.

Reference

Bond, J. (2021, January 27). *What Is Anime? A Brief History of Anime Genres, Culture, and Evolution*. The Daily Dot. https://www.dailydot.com/parsec/what-is-anime/

Kumar, S. (2021, August 9). *Silhouette Method — Better than Elbow Method to find Optimal Clusters*. Medium. https://towardsdatascience.com/silhouette-method-better-than-elbow-method-to-find-optimal-clusters-378d62ff6891