  
BACS2003 ARTIFICIAL INTELLIGENCE

**202301 Session, Year 2022/23**

**Assignment Documentation**

| **Full Name: Chiew Hong Kuang** | |
| --- | --- |
| **Student ID:** | |
| **Programme: Bachelor of Computer Science (Honours) in Software Engineering** | |
| **Tutorial Class: G7** | |
| **Project Title:** | |
| **Module In-Charged:** | |
| **Other team members’ data**   | **No** | **Student Name** | **Module In Charge** | | --- | --- | --- | | **1** | **Kong Zhi Lin** | **User based + Content based** | | **2** | **Tan Eng Lip** | **User based + Item based** | | |
| **Lecturer:** | **Tutor :** |

# **Introduction**

## Problem Background

In this modernization and technology development era, many people are seeking spiritual wealth rather than physical wealth. As a result, people all over the world are now seeking and enjoying a variety of entertainments, such as movies, sports, anime, and video games, in order to unwind after a period of frustration. (<https://doi.org/10.53730/ijhs.v6nS2.8231>)

In this project, we will discuss Japanese animation (also called anime) which is a novel culture that is rising in recent years. It has been proposed that there is a problem in finding a proper anime recommender system on the market, as this has caused many people to be unable to find the proper anime suitable for them and thus lead to a lack of publicity for this wonderful culture (<https://doi.org/10.1109/ISRITI51436.2020.9315363>). Fortunately, we have proposed a solution to this problem by recommending to users high-rated anime that is relevant to their viewing preferences.

## Objectives/Aims

In this project, our primary objective and goal is to improve the user experience by providing personalized anime recommendations that are tailored to their interests and needs based on the anime they have watched previously. We also provide accurate searches among thousands of search results due to the anime recommended system that we have proposed being capable of producing a relevant and appropriate list of anime for the users. This allows users to find what they are looking for quickly and easily. This eventually can increase the users’ productivity, such as discovering new anime they were not familiar with before and also increased user engagement in the anime streaming platform.

Moreover, we are targeting to promote the anime culture as well as provide a great impression of anime to the users who are fresh to anime via this system since we will produce a list of anime to recommend based on the ratings in our dataset. As a result, the satisfaction of the users will be increased and eventually enhanced by providing personalized and high-quality recommendations instead of searching blindly and effortlessly.

## Motivation

> (legal system → legal content, security and privacy)

First and foremost, a successful recommender anime system can lead to sales and revenue increases for anime streaming platforms or publishers. An anime recommender system also can help increase revenue by promoting content that users are more likely to watch or rent/buy. This can also allow the user to explore their interests in various genres and push the views of animes that will help promote newer or less popular content, which can lead to increased revenue from rentals or purchases.

Besides that, by providing personalized anime content recommendations, a recommender system can increase user retention by making it more likely that users will continue using the video/anime streaming platform. This is because the recommendation meets the user's interest and makes them more likely to continue viewing the anime they are interested in. A recommender system that suggests anime titles based on their interests and viewing habits can help to guide them towards shows that they may enjoy, making it easier for them to become fans of the medium.

Therefore, our system can reduce the barrier to entry until it can help to expose users to a wider range of anime titles and genres, which can promote diversity and inclusivity in the anime community. For people who are new to the anime medium, the vast number of titles and genres can be overwhelming and intimidating. The recommender system also can recommend the popular anime at the moment and this can help to expand the anime fanbase and promote greater awareness and appreciation for the art form.

As a consequence, this can help to support the anime industry and encourage the production of more high-quality anime titles in the future by the preferences of the anime fans. Additionally, our system can help all the anime related companies to gather data for better understanding on user preferences and behaviors, which can inform content production and acquisition strategies.

## Timeline/Milestone

Our project duration started from … until … (.. Days)

## 

# **Research Background**

## Background of the applications

*Provide detailed explanations of the background of the application, e.g. machine learning algorithms, chatbot development, recommender system, sentiment analytic applications, robotic processing automation applications, image processing applications, etc.*

A recommender system is a subset or type of artificial intelligence (AI) technology that provides personalized recommendations to users based on their information such as past behaviors, preferences, and patterns. The objective of a recommender system is to suggest items or content that the user is probably to be interested in, based on their previous interactions with the system. As we know, recommender systems have been used in a wide range of applications, from e-commerce and online advertising to music, movie and item recommendations.

There are several different types of recommender systems, including content-based filtering, collaborative filtering, and hybrid recommender systems. Content-based filtering involves recommending items that are similar to items that the user has previously interacted with, based on their attributes or features. On the other hand, collaborative filtering involves recommending items based on the preferences and behaviors of other users who have similar interests to the user in question. Hybrid recommender systems combine elements of both content-based and collaborative filtering.

Additionally, recommender systems rely on a variety of techniques and algorithms to make recommendations. These may include data mining, machine learning, and natural language processing. In order to be effective, recommender systems require a large amount of data about user behaviors and preferences, as well as the items being recommended.

## Analysis of selected tool with any other relevant tools

*Fill the table below and change the tools’ names. You may add more columns.*

| **Tools comparison** | **Remark** | **Jupyter Notebook**  **(Python 3)** | **Visual Studio Code** | **Google Docs** | **Microsoft Excel (2019)** |
| --- | --- | --- | --- | --- | --- |
| Type of license and open source license | State all types of license | Open source | Free for private and commercial use. | Free for Google users. | Microsoft Office License required |
| Year founded | When is this tool being introduced? | 2014 | 2015 | 2006 | 1985 (oldest version of Excel) |
| Founding company | Owner | Fernando Pérez and Brian Granger | Microsoft Corporation | Upstartle | Microsoft Corporation |
| License Pricing | Compare the prices if the license is used for development and business/commercialization | None | Free | Free | None  (pre-install in laptop, so free for us) |
| Supported features | What features that it offers? | - enables user to run interactive python code  - markdown language supported | - enable users to manage the code with the support of various extensions.  - enable users to execute code in various file types such as python, c, java, ipynb etc.  - Enable connection to github for version control and collaboration. | - Act as a collaborative tool for cooperative editing of documents online  - Can be shared, opened, modified, download by multiple users simultaneously.  - Grammarly-check  - Spelling-check  - Can retrieve the history in order to get all version | - inserting a pivot table  - sorting of tabulated data  - adding formulas to the sheet to perform calculation  - visualize the data |
| Common applications | In what areas this tool is usually used? | - understanding the dataset loaded in  - perform data science tasks | - manage and execute the code for the development process. | - Allow users to collaborate editing the same documentation and also manage the version of the document. | - perform data analysis (by add-ins in Excel)  - calculate the budget of the project |
| Customer support | How the customer support is given, e.g. proprietary, online community, etc. | Open-source community | Microsoft support | Google | Microsoft support |
| Limitations | The drawbacks of the software | Buffer on kernel will affect running of the kernel and requires to restart all the kernel and run again. | Difficult to manage plugins when used for different programming environments. | Requires internet connection to collaborate with other people or access to the document. | It is difficult to detect fraud/corruption |

## Justify why the selected tool is suitable

*Explain which tool is used for the development, and justify the suitability of the tool used in your project.*

>

Visual Studio Code is the application we used in order to create the recommendation algorithm to predict the suitable result or recommendation to the users. Visual studio code supports multiple programming languages and it also includes .ipynb format which is the jupyter notebook. In visual studio code, it allows the use of various plugins that enable the functions such as code completion, error detection, etc which will save a lot of time rather than trial and error. Besides that, it also supports the import of libraries such as pandas, string, re, time etc. It is also a crucial feature which allows us to directly use the well developed function rather than needing to write the code from scratch.

Besides that, visual studio code also allows the collaboration with other teammates by using the git to commit, push and pull the latest code into github and the other teammate can make changes to the source code without any conflict. Github also has the version control features which allows us to find back the older version code committed on github and trace back who has made the changes.

# **Methodology**

## Description of dataset

The source of the dataset comes from Kaggle which is a website that provides various open source dataset. Our dataset is the “Anime Recommendations Database” (<https://www.kaggle.com/datasets/CooperUnion/anime-recommendations-database?select=anime.csv>) provided by a user called “COOPERUNION”. There are two files in this dataset which are anime.csv and rating.csv. This data set contains information on user preference data from 73,516 users on 12,294 anime. Each user is able to add anime to their completed list and give it a rating and this data set is a compilation of those ratings. Inside the anime.csv had 7 columns of variables about the information of anime movies which are anime\_id, name, genre, type, episode, rating and members. There are 3 columns of variables of user rating which are user\_id, anime\_id and rating inside the rating.csv.

anime.csv

| Numbering Column | | Description |
| --- | --- | --- |
| anime\_id | | myanimelist.net's unique id identifying an anime. |

| Inputs/Features | Represent | Description |
| --- | --- | --- |
| name | Name | Full name of anime. |
| genre | Genre | Comma separated list of genres for this anime. |
| type | Type | Type of show such as movie, TV, OVA, etc |
| episodes | Episodes | The number of episodes in this show. (1 if movie). |
| rating | Rating | Average rating out of 10 for this anime. |
| members | Members | Number of community members that are in this anime's  "group". |

rating.csv

| Numbering Column | | Description |
| --- | --- | --- |
| user\_id | | Non identifiable randomly generated user id. |

| Inputs/Features | Represent | Description |
| --- | --- | --- |
| anime\_id | Anime\_id | The anime that this user has rated. |
| rating | Rating | Rating out of 10 this user has assigned (-1 if the user watched it but didn't assign a rating). |

newRating.csv

New defined dataset that is the same label column as the rating.csv file but the rating data is collected from the users to test on the accuracy of the anime recommended system.

| Numbering Column | | Description |
| --- | --- | --- |
| user\_id | | Non identifiable randomly generated user id. |

| Inputs/Features | Represent | Description |
| --- | --- | --- |
| anime\_id | Anime\_id | The anime that this user has rated. |
| rating | Rating | Rating out of 10 this user has assigned (-1 if the user watched it but didn't assign a rating). |

### **Data Preprocessing**

**Data Transformation**

There are some variables belonging to improper data types in the anime.csv such as the data type of name, genre, type and episodes belongs to object while the name, genre and type should belong to string type and the episodes belong to int64. Figure 3.1.1 below shows the original data type of the variables in the anime.csv while the Figure 3.1.2 shows the output after applying astype() method in pandas to convert the data type of the variables. Data transformation is necessary and important since the algorithm that is used for generating the recommended system is sensitive to the type and format of data.

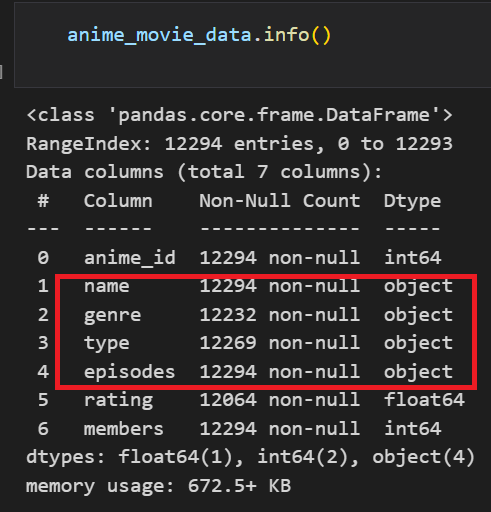


Figure 3.1.1 Before Data Transformation

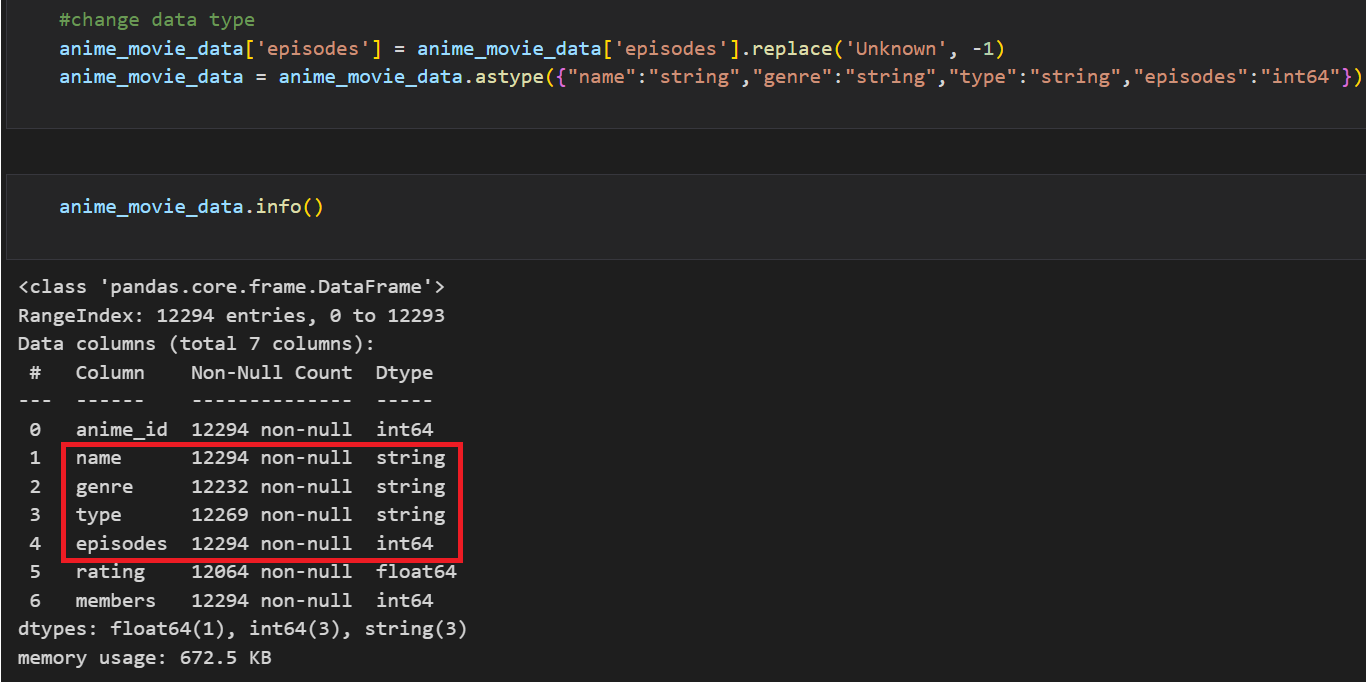


Figure 3.1.2 After Data Transformation by Using ‘astype()’

**Meaningful Variable Name**

1. rating

The column label of ‘rating’ appears at both the anime.csv and rating.csv but the meaning of ‘rating’ is different from each other. The ‘rating’ in the anime.csv means the average rating for each anime movie that is rated by the users while the ‘rating’ in the rating.csv means the user rating for each anime they had viewed. Therefore, there must be a modification on the variable name in order to reduce the confusion of the label. Figure 3.1.4 shows the original label of the anime.csv while Figure 3.1.5 shows the modified label of the anime.csv. Figure 3.1.6 shows the original label of the rating.csv and Figure 3.1.7 shows the modified label of the rating.csv.

‘rating’ in the anime.csv changes to ‘avg\_rating’

‘rating’ in the rating.csv changes to ‘user\_rating’



Figure 3.1.4 Original data frame of anime.csv with ‘rating’ label

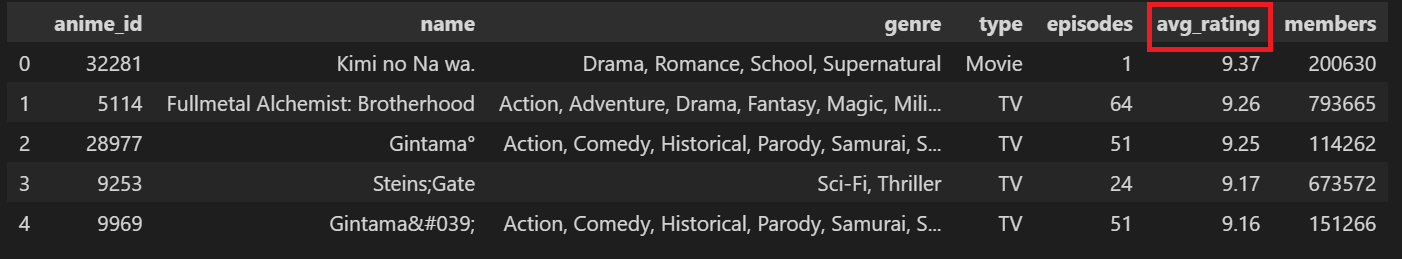


Figure 3.1.5 Modified data frame of anime.csv with ‘avg\_rating’ label

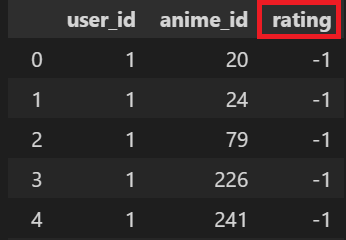


Figure 3.1.6 Original data frame of rating.csv with ‘rating’ label

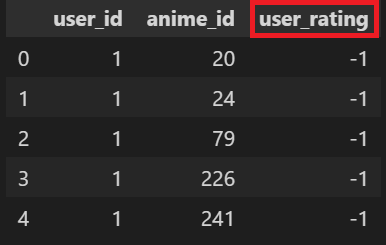
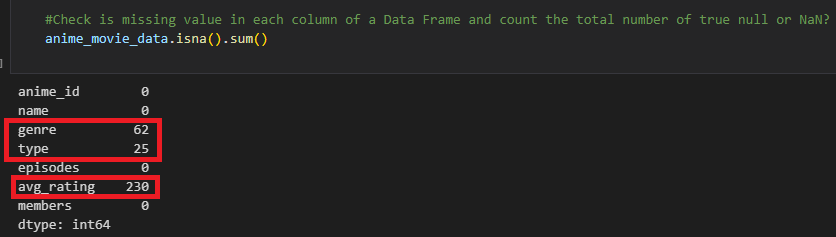


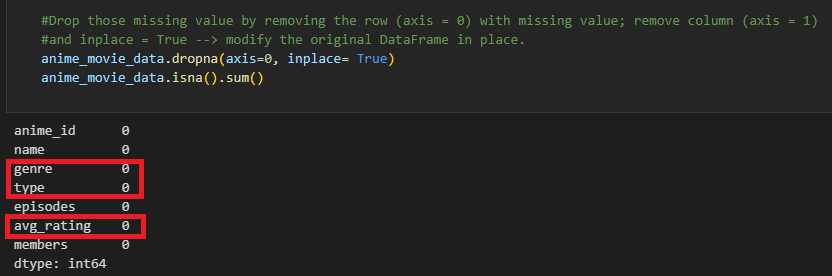
Figure 3.1.6 Original data frame of rating.csv with ‘rating’ label

**Removing Null Value**

Ensure there is no missing value of the data in the data frame by using the isna().sum() method in the pandas. dropna() method is used to remove all the missing values in the data frame.



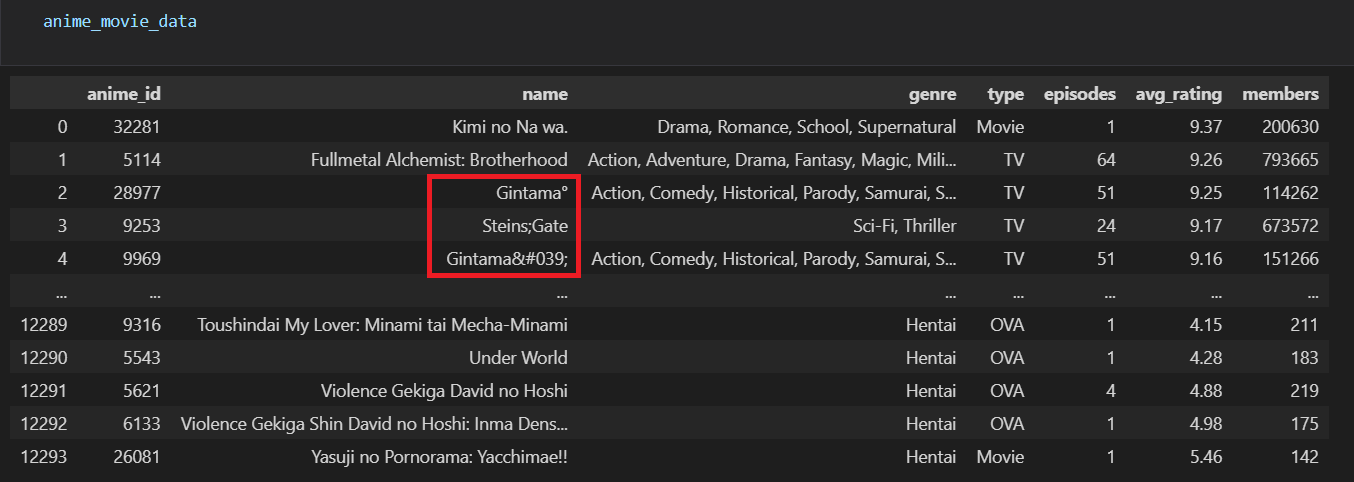
Before removing the missing value



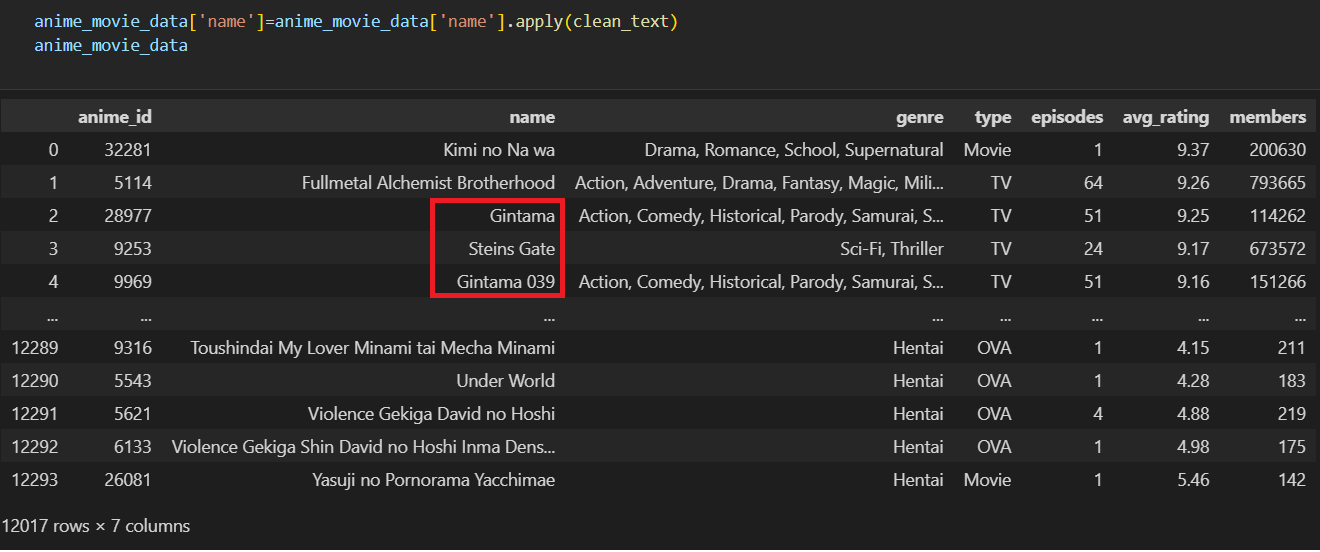
After removing the missing value

**Clean Text**

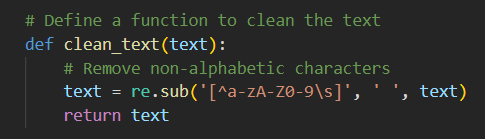
It is found that the anime movie contains special characters which will increase the difficulty of reading and identify the name. By using the clean\_text() function we can remove all those special characters in the name to make it readable and understandable.



Before using clean\_test() function



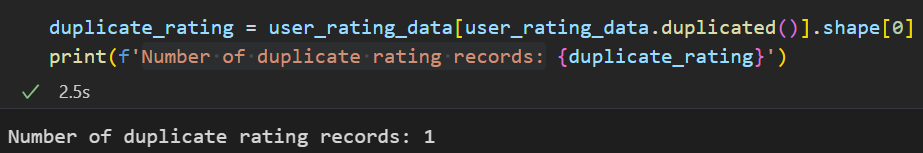
After using clean\_test() function



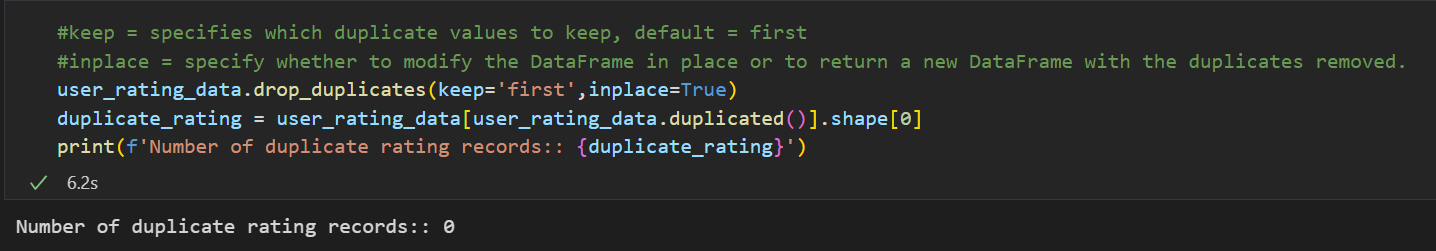
clean\_text function

**Remove Duplicate Records**

There should not be any duplicate records occur in the anime.csv and rating.csv since it will make the recommender system become not accuracy and waste resources which will slow down the system’s performance. It is found that the rating.csv had 1 duplicate records by using duplicated() method to figure out and using the drop\_duplicates() method to remove the duplicate records.



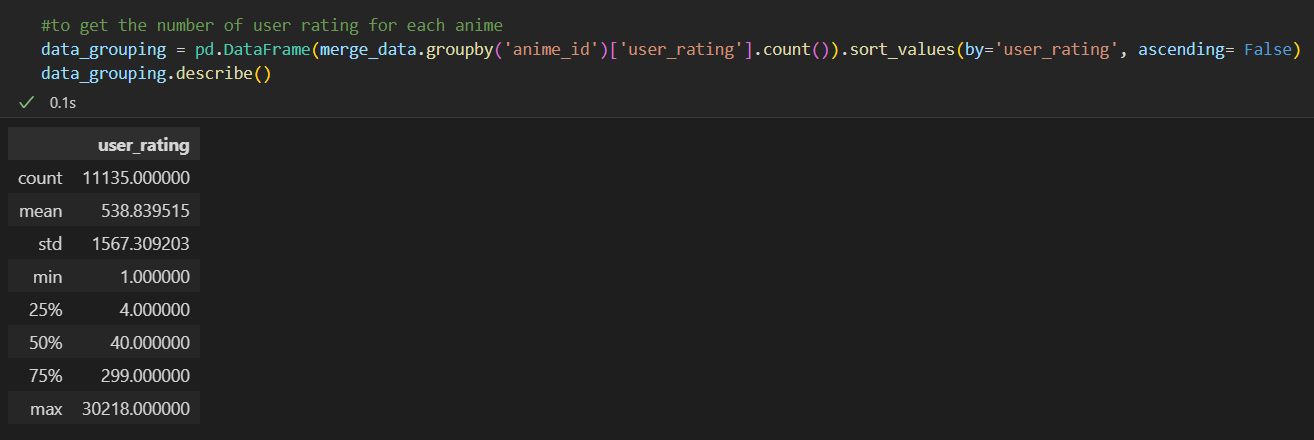
Before removing duplicate rating records



After removing duplicate rating records by using drop\_duplicates() function

**Data Trimming**

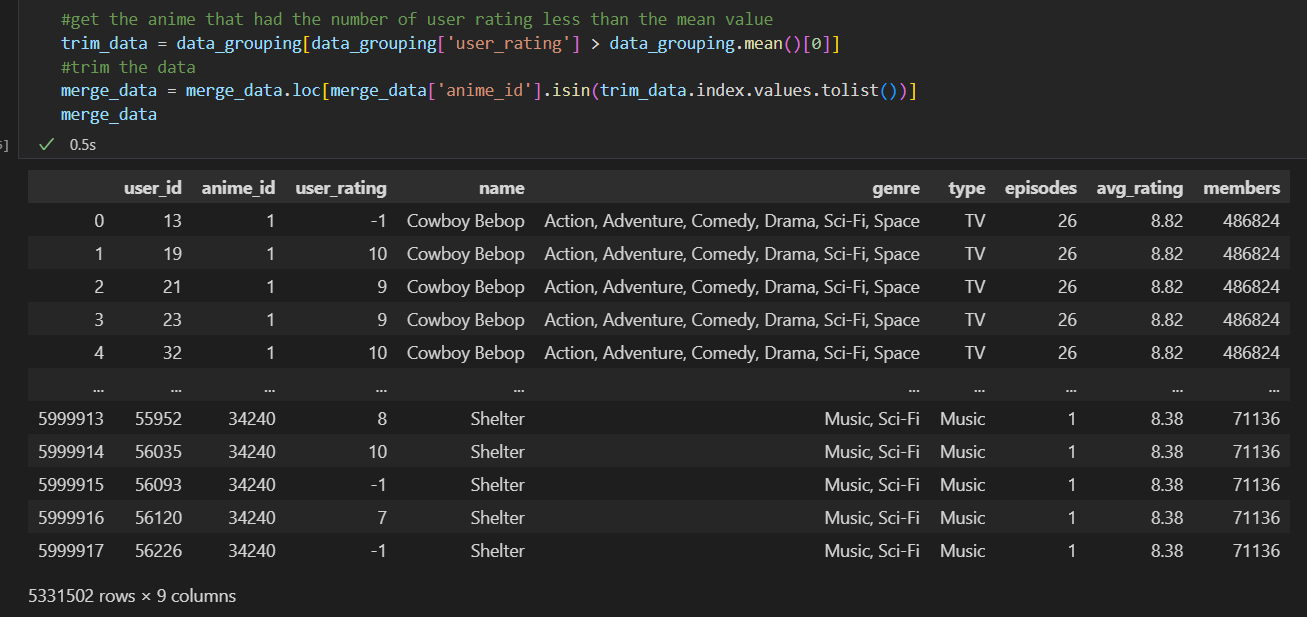
Data trimming is used to remove extreme ratings given by users that do not accurately represent their preferences in order to increase the accuracy of the recommended system. With the help of mean() after we group the anime\_id and the user\_rating by the groupby(), we can calculate the mean value omong all these user rating values of the anime. Those values of user rating is less than the mean value will be trimmed.



Group the anime\_id and user\_rating by using the groupby()

&

Use describe() to display the mean, standard deviation, minimum value and so on



Trim data for those user rating value is less than the mean

## Applications of the algorithm(s)

The algorithm that was used in this project is the combination of content-based filtering and the user-based collaborative filtering in order to provide anime recommendations to the users. The system will use the information about the movies such as genre, name, type and so on for the content-based filtering while the user-based filtering will be using the information about the users’ interactions with the movies such as the user ratings.

**Content-based filtering**

Content-based filtering will use the content features of movies including the name of the movie, genre of movie, type of movie and so on to recommend similar movies to the users. In this filtering, the tf-idf ( term frequency-inverse document frequency) matrix is constructed to represents each document in the collection as a vector of tf-idf values, the cosine similarity matrix is also constructed to represents the similarity between each pair of documents in the collection based on their tf-idf vectors and compute the cosine similarity score to each anime. The anime with the highest similarity score will be recommended to the users. Figure 3.2.1 shows the histogram of the frequency of cosine similarity scores from the 10 highest scores of anime.

Figure 3.2.2 shows the result of anime recommendation with the cosine similarity scores by using the content-based filtering.

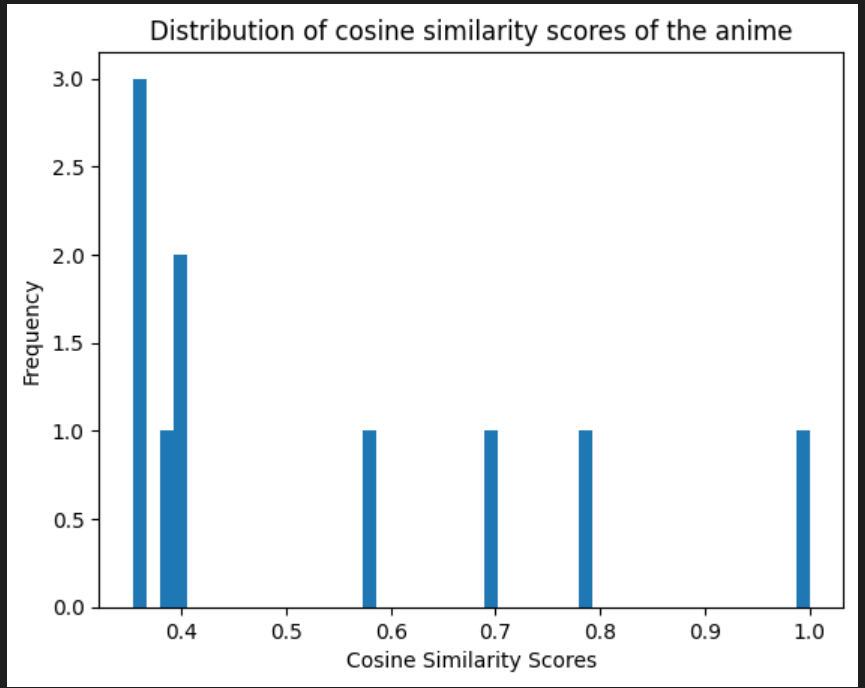


Figure 3.2.1 Histogram of frequency of the cosine similarity scores of anime

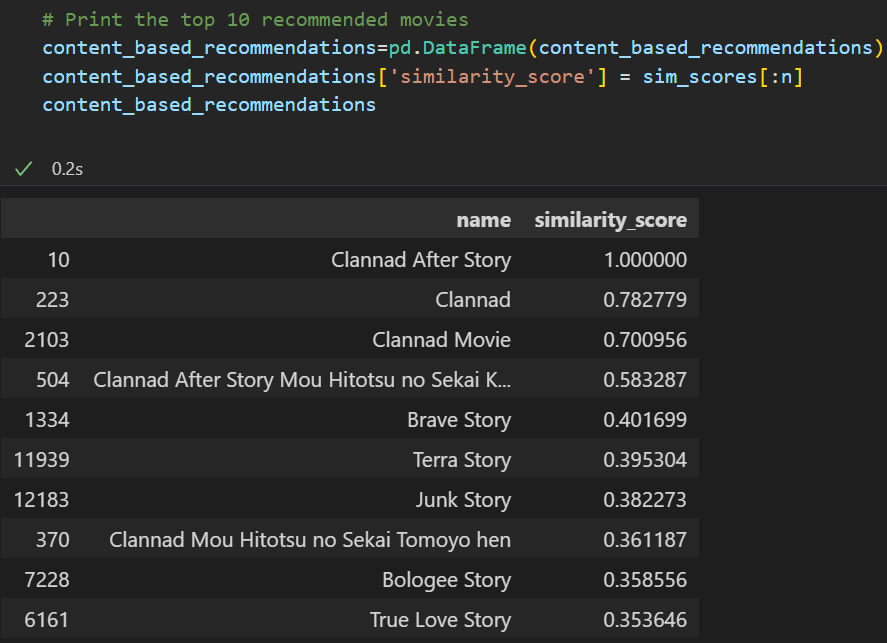


Figure 3.2.2 Result of content-based filtering

**User-based collaborative filtering**

User-based collaborative filtering makes use of the preferences of similar users to generate recommendations which means the system identifies users who have similar preferences, view histories to the current user, and then recommends the anime. In this filtering, Pearson Correlation Coefficient is used to determine the relationship between one user and anothers. The anime with the highest correlation between the selected users will be recommended to the users. Figure 3.2.3 shows the histogram of the frequency of correlation scores from the 10 highest scores of the users.

Figure 3.2.4 shows the result of anime recommendation with the correlation scores by using the user-based filtering.

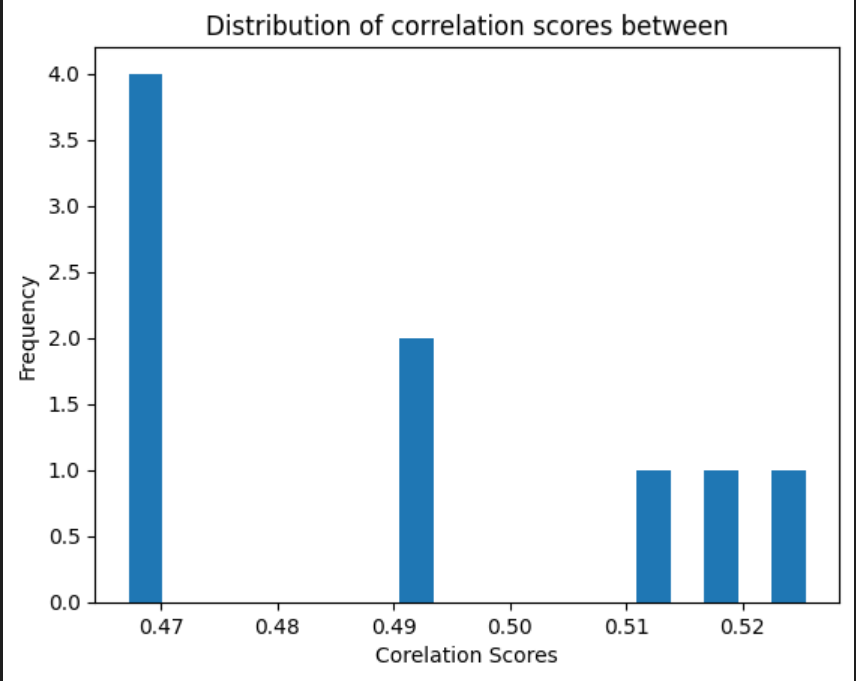


Figure 3.2.1 Histogram of frequency of the correlation scores of user

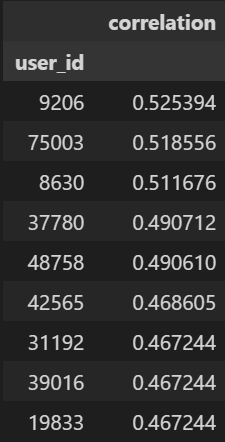
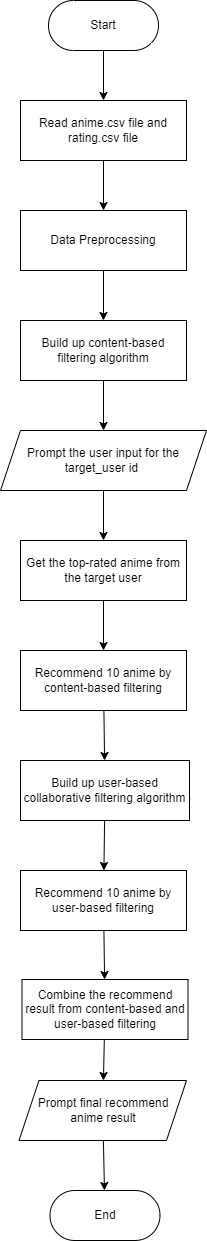


Figure 3.2.2 Result of user-based filtering

## System flowchart/activity diagram

**

## Proposed test plan/hypothesis

**Hpothesis**

Hypothesis that I had made for this project (combination of content-based filtering and the user-based filtering) is that the final recommended anime will have more than 80% accuracy.

**Test plan**

Since there does not have a discrete or systematic measurement to test on the accuracy of the recommended system, but it highly depends on the user who provides the data and the system recommend to the users as well as to get the satisfaction of the users. Therefore, in this case, a few people are going to help test the accuracy of this project by receiving their viewed anime and the rating about those anime. Then, we recommend them some anime via the 2 filtering algorithm and get the feedback of their satisfaction about the recommendation.

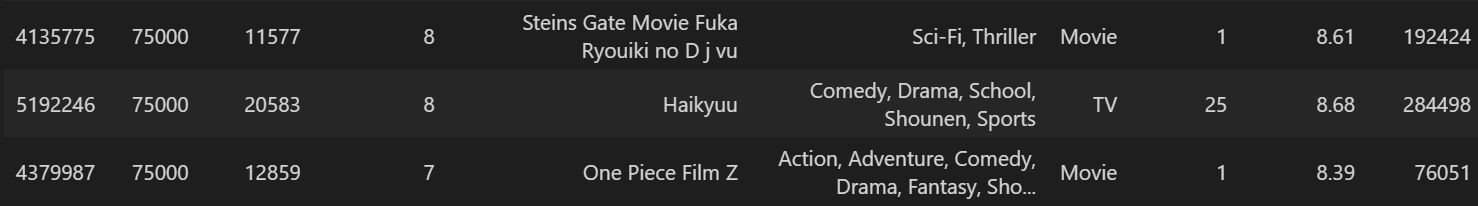
# **Result**

## Results

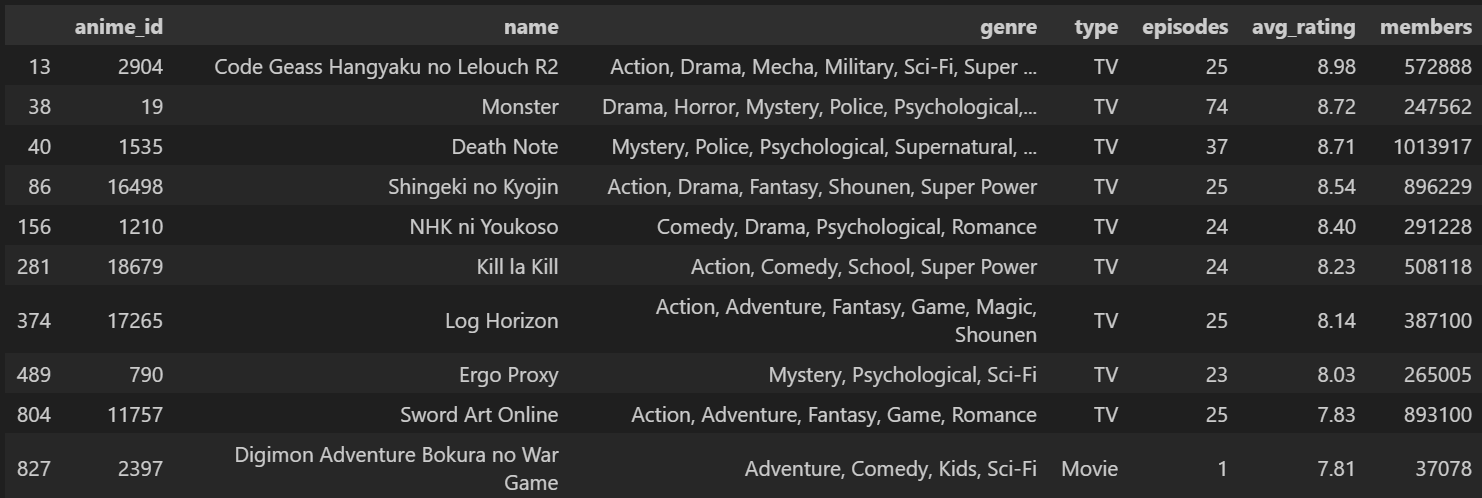
The user\_id starts from 75000 until 75005 is the users that used to test on the recommended system.

75000’s user rating for the anime he had viewed:





Recommended anime to 75000’s user:

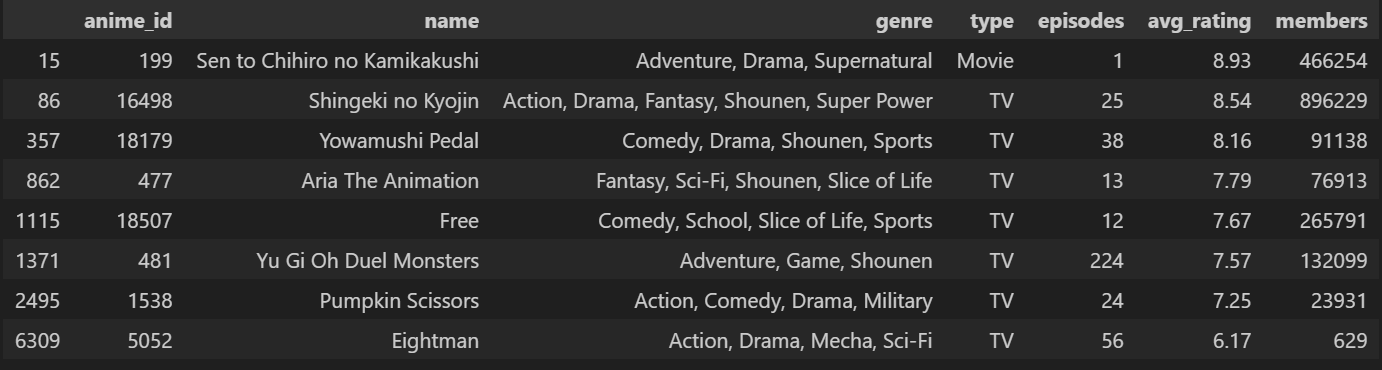


There are 8 out of 10 anime recommendations that users are interested in and satisfied with the recommended result. The 2 anime the users are not interested in are ‘Monster’ and ‘Ergo Proxy’. Percentage of accuracy of this recommended system is 80%.

75001’s user rating for the anime he had viewed:

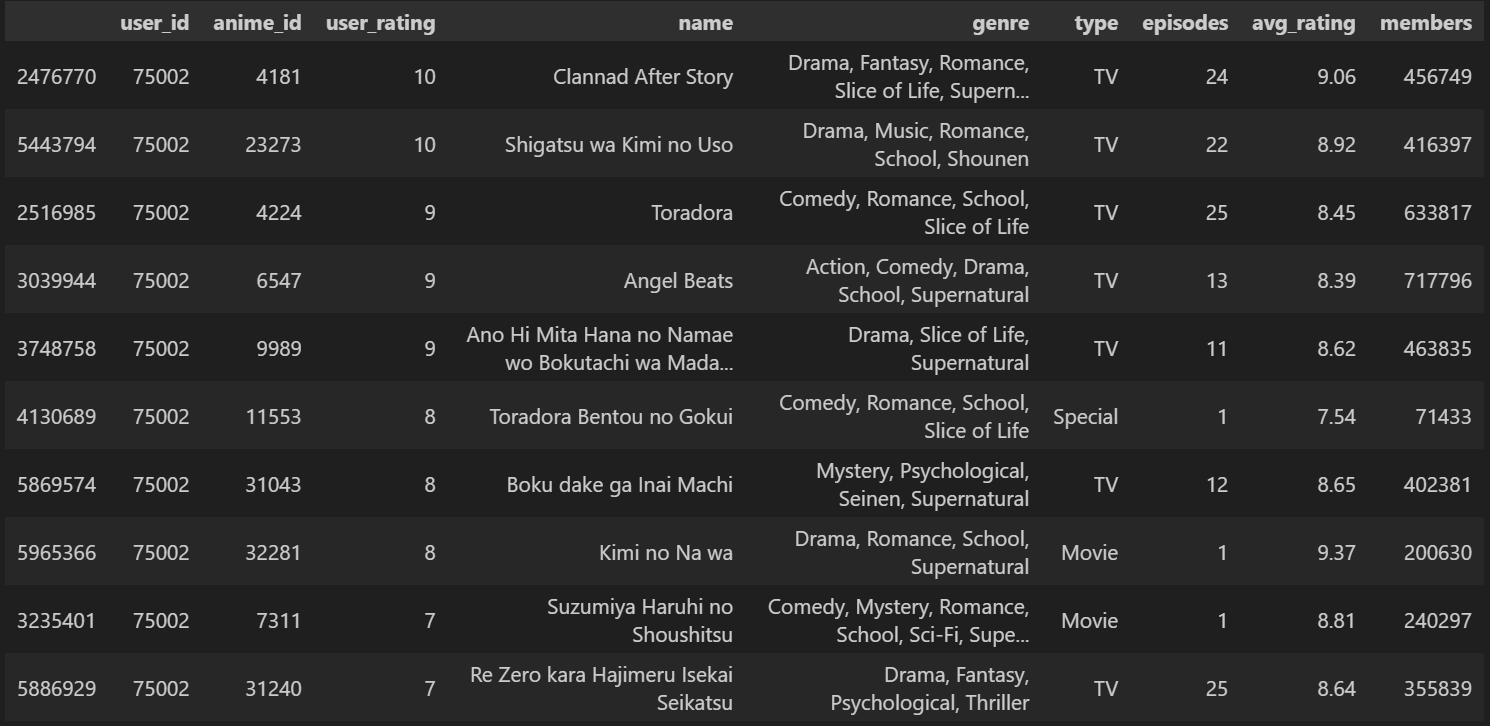


Recommended anime to 75001’s user:

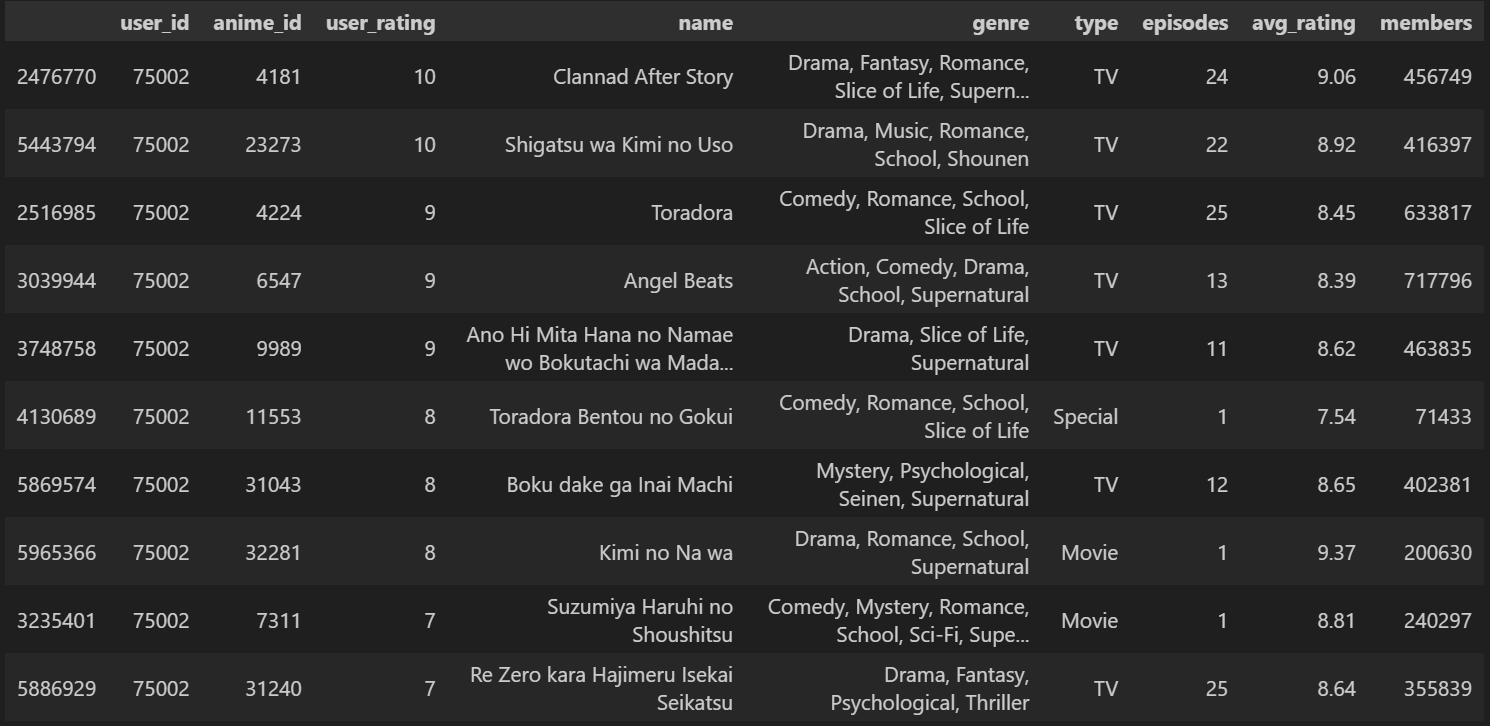


There are 5 out of 8 anime recommendations that users are interested in and satisfied with the recommended result. The 3 anime that the users are not interested in are ‘Yowamushi Pedal’, ‘Free’ and ‘Pumpkin Scissors’ . Percentage of accuracy of this recommended system is 62.5%.

75002’s user rating for the anime he had viewed:



Recommended anime to 75002’s user:

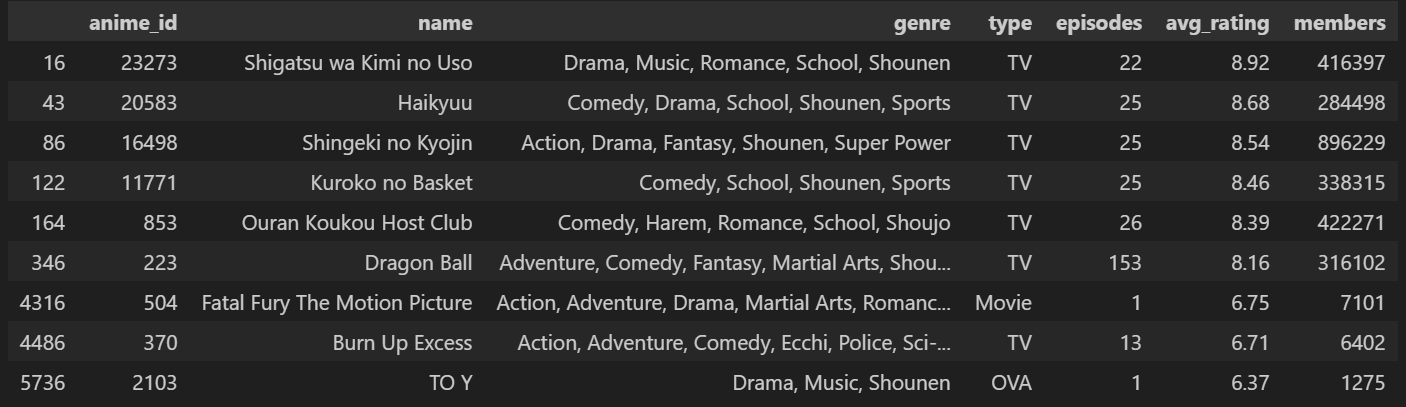


There are 10 out of 10 anime recommendations that users are interested in and satisfied with the recommended result. The user is interested in all of the anime that are recommended. Percentage of accuracy of this recommended system is 100%.

75003’s user rating for the anime he had viewed:



Recommended anime to 75003’s user:



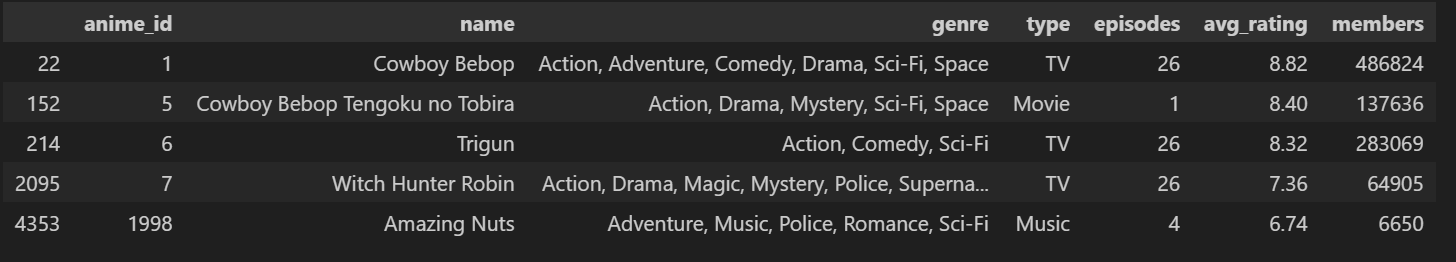
There are 8 out of 9 anime recommendations that users are interested in and satisfied with the recommended result. The anime that the users are not interested in are ‘To Y’. Percentage of accuracy of this recommended system is 88.8%.

75004’s user rating for the anime he had viewed:





Recommended anime to 75004’s user:

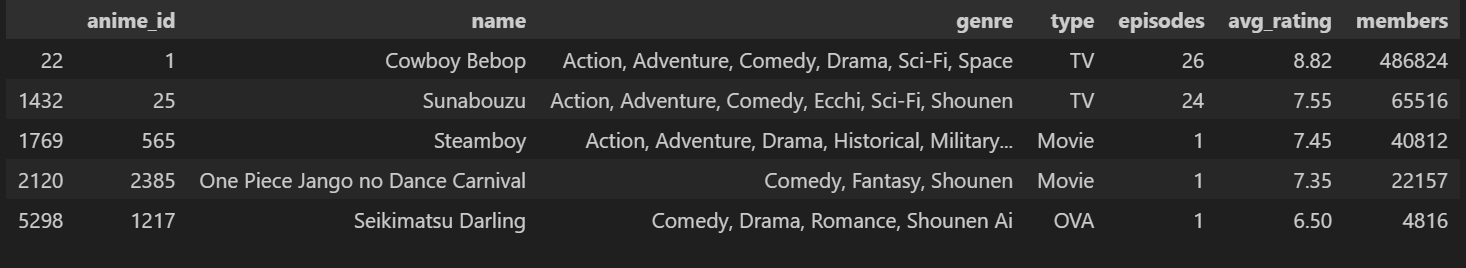


There are 4 out of 5 anime recommendations that users are interested in and satisfied with the recommended result. The anime that the users are not interested in are ‘Trigun’, ‘Free’ and ‘Pumpkin Scissors’ . Percentage of accuracy of this recommended system is 80%.

75005’s user rating for the anime he had viewed:



Recommended anime to 75005’s user:



There are 5 out of 5 anime recommendations that users are interested in and satisfied with the recommended result. The users are interested in all the anime that the system recommended. Percentage of accuracy of this recommended system is 100%.

## Discussion/Interpretation

The percentage of average accuracy of this anime recommended system with the combination of content-based filtering and user-based collaborative filtering is 85.22%. The formula to calculate the percentage of average accuracy is:

**Sum of the accuracy of recommendation for each user / number of users test the accuracy**

Calculation: (80% + 62.5% + 100% + 88.8% + 80% + 100%) / 6 = 85.22%

Therefore the hypothesis is accepted due to the percentage of average accuracy of this anime recommended system being over 80%.

# **Discussion and Conclusion**

## Achievements

This anime recommended system which contains 2 combination of content-based filtering and the user-based filtering

## Limitations and Future Works

## *Discuss the limitations of the project and what improvements can be done in the future*

>

## 

# **Reference & Source**

## *Provide the sources of the dataset and tool(s) used for development*

## *List the articles or other references you have cited in the text using the Harvard Referencing system.*

## 

# 

# 

# 