LONDON
METROPOLITAN
UNIVERSITY

islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

CU6051NI - Artificial Intelligence

Assessment Weightage & Type

25% Individual Coursework

Year and Semester

2020-21 Autumn

Student Name: Suyogya Luitel

London Met ID: 19031784

College ID: np01cp4a190035

Assignment Due Date: 22nd December 2021

Assignment Submission Date: 22nd December 2021

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Table of Contents

Table of Figures:

Table of Tables

1. Introduction

1.1. Project Domain Introduction

The chosen topic for the coursework is an anime recommendation system. Anime is hand-drawn or computer animation videos originating from Japan. Anime has been growing as a digital entertainment media not only in Japan but throughout the globe. According to a market analysis report by Grand View Research based on historical data from 2018 to 2019, the global anime market size was valued at USD 23.56 billion in 2020 and is expected to expand at a compound annual growth rate (CAGR) of 9.5% over the forecast period (2021-2028) (Grand View Research, 2021). Taking reference of the popular website named My Anime List keeping track of and rating most of the anime produced to date, there have been over 16,000 animes with multiple episodes, covering over 20 genres. However, that is not the end of anime production.

1.2. Real-World Application of Project

As time moves forward, more and more anime will be added to the pool of choices. As such, it can be quite a daunting experience to make a choice. For the service provider as well, having a group of consumers wasting their time trying to make a choice instead of using that time to consume the services provided turns out to be costly. The problem would be resolved if there was a system in place to hand out personalized recommendations to each user based on the vast amount of user data collected by the service providers. Recommendation systems do just that. A recommendation system helps users to determine which item they might be interested to use. In the scenario described above, a recommendation system enables the consumers to make quick decisions thus minimizing the wasted time. This also means less operating expense for the service provider.

1.3. Project Overview

For this project, the dataset to use is extracted from the My Anime List website and available on the data science site Kaggle. The dataset contains data about the site users, their reviews as well as data about the animes. The users give a rating score to an anime on a scale of one to ten. For this project, we propose the use of a recommendation system using collaborative filtering. This approach is proposed to measure the similarity between shows, users and helps to predict which anime a user may enjoy.

2. Background

2.1. Brief History of Recommendation Systems

The recommendation system was initially mentioned in a technical report by Jussi Karlgren at Columbia University as "digital bookshelf" in 1990. According to Jussi, the paper was rejected at the 1990 INTERACT due to user privacy and integrity issues (Karlgren, 2017). He later completed his work in 1994 when he worked at SICS (Karlgren, 1994). Since then, several types of recommendation systems have come to light. Some of the popular recommendation systems are:

i. Collaborative recommender systems
ii. Content-based recommender systems
iii. Hybrid recommender systems
iv. Session-based recommender systems

2.2. Brief Explanation of Collaborative Filtering Systems

Considering the nature of the problem, a collaborative recommender system seems to be the most suitable approach. A collaborative recommendation system works around a core assumption that users who liked something in the past will always like similar kinds of things. According to google, collaborative filtering uses similarities between users and items simultaneously to provide recommendations (Google, 2021).

According to Shuyu Luo, a collaborative recommender system can be either user-based or item-based. The user-based system calculates the similarity of a user with all other users. The top n number of similar users are selected and their weighted average is taken, where the user similarity is taken as the weight. While rating an item, some people tend to give high ratings in general, while some give rather lower ratings in general. To reduce this bias, each user’s average rating of all items is subtracted while calculating the weighted average and added back for its target user. (Luo, 2018)

2.3. Considered Algorithms for Calculating Similarity

Some popular algorithms for calculating similarity are:

i. Euclidian Distance

The Euclidean distance is the straight-line distance between two points (Rosalind, 2021). If (x_1, y_1) and (x_2, y_2) represent two points in Euclidian space, the Euclidean distance formula is: $((x_2 - x_1)^2 + (y_2 - y_1)^2)^{1/2}$.

This algorithm can be used to find similarities between users or items by assuming that the ratings are vertices in a Euclidian space where items and users are the dimensions. The users or items whose ratings have the least straight-line distance are the most similar to one another. While this is a straightforward analogy, it starts making lesser sense as the number of dimensions keeps increasing.

ii. Manhattan Distance

The Manhattan distance, also called the Taxicab distance or the City Block distance, calculates the distance between two real-valued vectors (Brownlee, 2020). If (x_1, y_1) and (x_2, y_2) represent two points in space, the Manhattan distance formula is: $|x_2 - x_1| + |y_2 - y_1|$. This metric for measuring similarity generally works best if the points are arranged in the form of a grid. It can also be easily generalized in higher dimensions.

iii. Pearson coefficient

Pearson coefficient is a measure that quantifies the strength of the linear relationship between two variables in a correlation analysis (Statistics Solutions, 2021). Instead of the previous approaches that calculated a similarity score based on spatial distances, this approach uses a correlation scenario to predict the user's potential rating.

2.4. Pros and cons of Collaborative Approach

Some pros of using a collaborative approach for a recommendation system are:

i. The system does not require any domain knowledge.
ii. The system can help users discover interests in new sectors as well based on the preference of a similar group.
iii. The system is simple and runs based on a user feedback matrix and does not require further context information.

Some cons of using a collaborative approach for a recommendation system are:

i. The system cannot handle a cold start scenario where user preference or feedback is not available.
ii. Adding additional features or side features is a difficult task to perform.

2.5. Related Works:

As elaborated above, recommendation systems are not a new concept and they have been around since 1994. Besides Jussi Karlgren, several other researchers have researched the topic using collaborative, content-based, and hybrid approaches.

In their paper published in the 2016 International Conference on Next Generation Intelligent Systems (ICNGIS), Arpan V Dev and Anuraj Mohan proposed a solution for recommending items for big data applications. They made use of a user-based collaborative filtering algorithm and MapReduce framework. Their system was able to reduce the computation that was conventionally required while working with recommendation systems on big data. (Dev & Mohan, 2016)

In an article published in the 2015 International Journal of Computer Applications, Kumar proposed a film recommendation system. The proposed system uses collaborative filters that focus on the ratings given by users to provide recommendations. The system sorts the recommended movie list based on the movie ratings given by users using the K-means algorithm. The system helps users to find the movies of their choice based on the movie experience of other users. (Kumar, et al., 2015)

In a paper published in the 2017 3rd International Conference on Electrical Information and Communication Technology (EICT), Shahjalal and his team implement a recommender system that combines both collaborative and user-based filtering. Their approach saved the time-consuming, expensive and difficult nature of content-based recommender as well as saved the user from a lot of hassles like filling up a long survey form. (Shahjalal, et al., 2017)

3. Proposed Solution

3.1. Problem Formulation

The proposed solution to the problem is a collaborative recommendation System. The proposed solution can be broken down to:

To clearly understand the problem, a table of anime ratings is extracted from the given dataset of different anime names as the rows and their corresponding users as the columns. For example:

Animes	User A	User B	User C	User D	User E
X	5	?	6	7	10
Y	6	8	?	5	5
Z	?	6	1	2	?

Table 1: Anime ratings by users

In the table above, user A gives ratings of 5 and 6 to animes X and Y. This means that the user enjoyed anime Y more than anime X. The more the rating number, the more the user has enjoyed the anime. The user does not have a rating for anime Z. This means that the user has not watched anime Z. The maximum rating a user can give is 10 and the minimum is 1. Similarly, the table also shows ratings for anime Y and Z provided by other users, B, C, D, and E. The system will be built to recommend which anime a user should use based on historical data. The anime has a rating that surpasses the average threshold among the similar group will be recommended to the user.

3.2. Flowchart of the System:

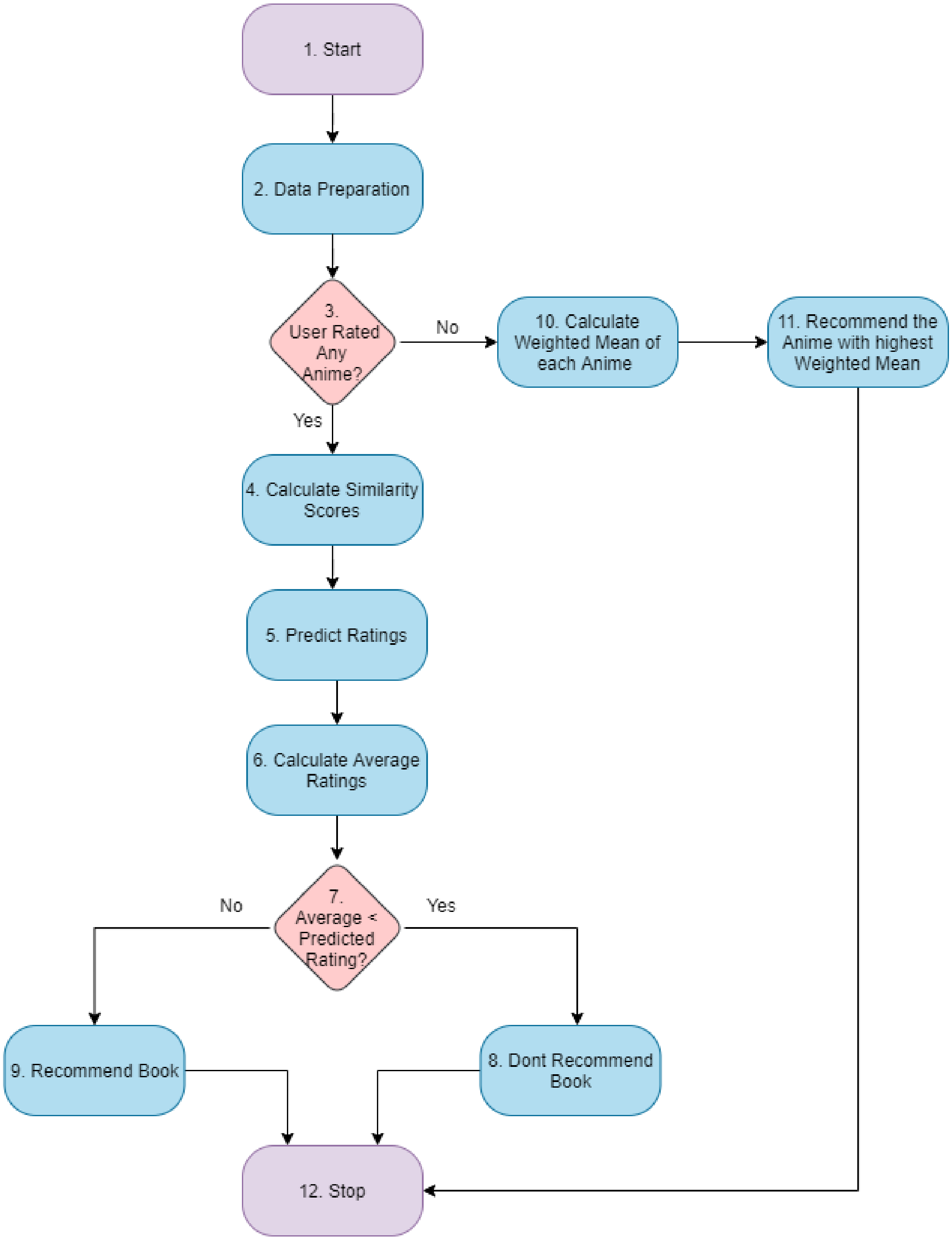
1. Start
2. Data Preparation
3. User Rated Any Anime?
No
10. Calculate Weighted Mean of each Anime
11. Recommend the Anime with highest Weighted Mean
Yes
4. Calculate Similarity Scores
5. Predict Ratings
6. Calculate Average Ratings
7. Average < Predicted Rating?
No
Yes
9. Recommend Book
8. Dont Recommend Book
12. Stop

Figure 1: Flowchart of System

The diagram presented above elaborates the flow of the proposed system.

3.2.1. Data Preparation

Data preparation is the step where useful information like anime details, user details, and ratings are extracted and divided.

3.2.2. Check Rating

This step involves checking if the user has rated any anime yet or not. Such checking is done to determine if the user is a new user with no reference data or not.

3.2.3. Calculate Similarity Score

This step involves calculating the similarity score of the user with all other users. The considered means for calculating such similarity are Euclidian Distance, Manhattan Distance, and Pearson coefficient as explained above in the background section.

3.2.4. Predict Rating

This step involves predicting the ratings of unwatched animes by the user.

3.2.5. Calculate Average Rating

This step involves determining a rating threshold, which in this case is done by calculating the average rating.

3.2.6. Check Predicted Rating

This step involves checking if the predicted anime passes the threshold. If it does, it is recommended to the user. Otherwise, it is not recommended to the user.

3.2.7. Calculate weighted mean

This step is carried out in the case of a cold start, where the user has not rated a single anime. The weighted mean of every anime rating is calculated to somewhat equalize the rating frequency of different animes. The anime with the highest weighted mean is then recommended to the user.

4. Conclusion

The anime recommendation system recommended is a solution based only on user watch history. This simple system can measure similarities between users and help predict how much the user may enjoy a certain anime. The solution also addresses the cold start problem that collaborative systems suffer from and provides a suitable alternative solution. The solution provided should in theory work efficiently for smaller datasets, but using it on larger datasets such as big data will require tremendous processing power. The system can be used not only for recommending anime but for anything involving user rating and a product. The system can also be used for grouping similar animes based on user reviews with a few changes.

The system only requires user feedback on the anime. Hence, it can be easily integrated into any anime site or anime rating site. The system helps users to find new interests, recommends them animes to watch without having to go through the search bar. Thus, the time a user spends on the anime site can be efficiently utilized by viewing the content rather than searching for content to watch. This can also be beneficial for the site hosts or owners.

As of now, the project is merely only a theoretical solution. In the future, the project is to be built using appropriate tools and frameworks such as Scikit Learn, Pandas, NumPy, etc. While building the system, the algorithm to use for calculating user similarity is also to be considered based on validation results. The system is a simple system that requires only user feedbacks on anime to work. There are several ways to build a recommendation system that may end up giving better results. Further research on optimizing the proposed solution to the problem also needs to be done.

5. Bibliography

Brownlee, J., 2020. *4 Distance Measures for Machine Learning.* [Online]
Available at: https://machinelearningmastery.com/distance-measures-for-machine-learning/
[Accessed 19 December 2021].

Dev, A. V. & Mohan, A., 2016. *Recommendation system for big data applications based on set similarity of user preferences,* s.l.: IEEE.

Google, 2021. *Collaborative filtering.* [Online]
Available at: https://developers.google.com/machine-learning/recommendation/collaborative/basics
[Accessed 17 December 2021].

Grand View Research, 2021. *Anime Market Size & Share, Industry Report, 2021-2028.* [Online]
Available at: https://www.grandviewresearch.com/industry-analysis/anime-market#:~:text=b.-,The%20global%20anime%20market%20size%20was%20valued%20at%20USD%2023.56,USD%2048.03%20billion%20by%202028.
[Accessed 12 December 2021].

Karlgren, J., 1994. *Newsgroup Clustering Based On User Behaviour - A Recommendation Algebra,* s.l.: SICS.

Karlgren, J., 2017. *A digital bookshelf: original work on recommender systems.* [Online]
Available at: https://jussikarlgren.wordpress.com/2017/10/01/a-digital-bookshelf-original-work-on-recommender-systems/
[Accessed 15 December 2021].

Kumar, M., Yadav, D., Singh, A. & Gupta, V. K., 2015. *A Movie Recommender System: MOVREC,* s.l.: International Journal of Computer Applications.

Luo, S., 2018. *Introduction to Recommender System.* [Online]
Available at: https://towardsdatascience.com/intro-to-recommender-system-collaborative-filtering-64a238194a26
[Accessed 17 December 2021].

Rosalind, 2021. *Euclidean distance.* [Online]
Available at: https://rosalind.info/glossary/euclidean-distance/#:~:text=The%20Euclidean%20distance%20between%20two,representing%20distance%20between%20two%20points.
[Accessed 19 December 2021].

Shahjalal, M. A., Ahmad, Z., Arefin, M. S. & Hossain, M. R. T., 2017. *A user rating based collaborative filtering approach to predict movie preferences,* Khulna: IEEE.

Statistics Solutions, 2021. *Pearson's Correlation Coefficient.* [Online]
Available at: https://www.statisticssolutions.com/free-resources/directory-of-statistical-

analyses/pearsons-correlation-coefficient/
[Accessed 20 December 2021].