IJARCCE
ISSN (O) 2278-1021, ISSN (P) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 10, Issue 11, November 2021
DOI: 10.17148/IJARCCE.2021.101127

RECOMMENDER SYSTEM

VANSH ARORA

Student, IT, Maharaja Agrasen Institute of Technology, Delhi, India

Abstract : This paper is based on a method of recommending movies to the user after assuming the priorities set/entered by the user. We now live in what some call the "era of abundance". For any given product, there are sometimes thousands of options to choose from. Think of the examples above: streaming videos, social networking, online shopping; the list goes on. Recommender systems help to personalize a platform and help the user find something they like. The algorithm used segregates the list of movies from the dataset according to the inputs provided by user and finally displays the list of the recommended movies. The work in this project focuses on user-based collaborative filtering algorithm which is implemented in java. A bright feature of allowing the user to rate movies has also been provided in the system.

I. INTRODUCTION

Recommender System is a system that seeks to predict or filter preferences according to the user's choices. Recommender systems are utilized in a variety of areas including movies, music, news, books, research articles, search queries, social tags, and products in general.Recommender systems encompass a class of techniques and algorithms that can suggest relevant items to users. They predict future behavior based on past data through a multitude of techniques.
We now live in what some call the "era of abundance". For any given product, there are sometimes thousands of options to choose from. Think of the examples above: streaming videos, social networking, online shopping; the list goes on. Recommender systems help to personalize a platform and help the user find something they like.
The easiest and simplest way to do this is to recommend the most popular items. However, to really enhance the user experience through personalized recommendations, we need dedicated recommender systems.
From a business standpoint, the more relevant products a user finds on the platform, the higher their engagement. This often results in increased revenue for the platform itself. Various sources say that as much as 35–40% of tech giants' revenue comes from recommendations alone.

II. BACKGROUND

A. TYPES OF RECOMMENDER SYSTEMS
Machine learning algorithms in recommender systems typically fit into two categories:
1) Content-based systems.
2) Collaborative filtering systems.
Modern recommender systems combine both approaches.
1) Content-Based Movie Recommendation Systems :
Content-based methods are based on the similarity of movie attributes. Using this type of recommender system, if a user watches one movie, similar movies are recommended. For example, if a user watches a comedy movie starring Adam Sandler, the system will recommend them movies in the same genre or starring the same actor, or both. With this in mind, the input for building a content-based recommender system is movie attributes.

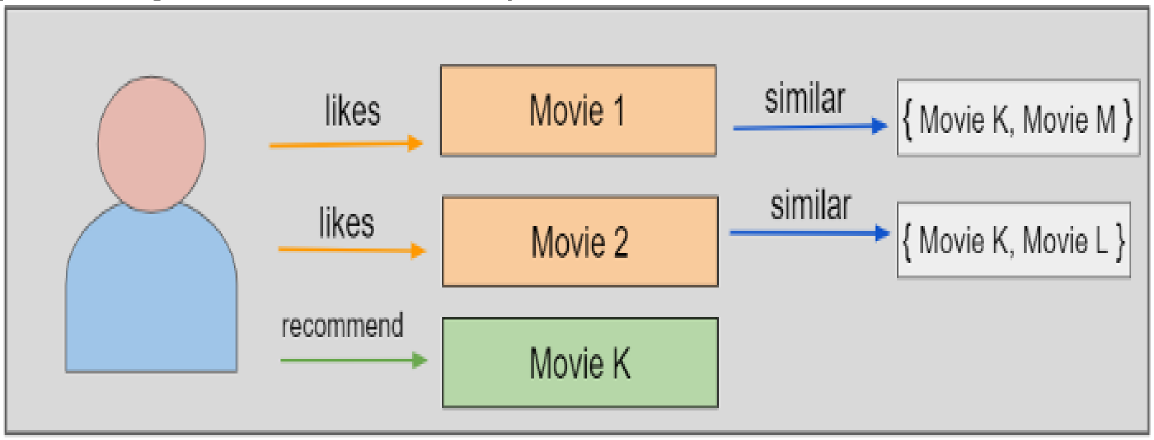


Fig. 1

IJARCCE
ISSN (O) 2278-1021, ISSN (P) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 10, Issue 11, November 2021
DOI: 10.17148/IJARCCE.2021.101127

2) Collaborative Filtering Movie Recommendation Systems :
With collaborative filtering, the system is based on past interactions between users and movies. With this in mind, the input for a collaborative filtering system is made up of past data of user interactions with the movies they watch.
For example, if user A watches M1, M2, and M3, and user B watches M1, M3, M4, we recommend M1 and M3 to a similar user C. You can see how this looks in the figure below for clearer reference.
This data is stored in a matrix called the user-movie interactions matrix, where the rows are the users and the columns are the movies.

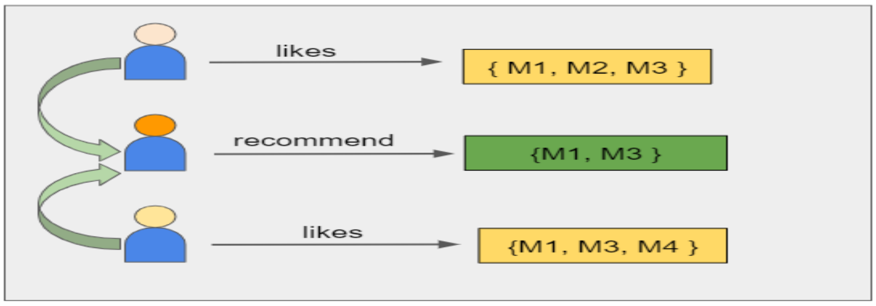


Fig. 2

III. PROPOSED METHODOLOGY

• Gather Data.
• Parse the dataset from Movies.java. Map the movie-id as key and movie name as value.
• In User.java, map user-id as key and user rating as value.
• Take the rating input from user.
• Map average rating of each user.
• Make a similarity array of each user in the dataset using Pearson correlation: - sim(i,j) = numerator / (sqrt(userDenominator^2) * sqrt(otherUserDenominator^2)) numerator = sum((r(u,i) - r(u)) * (r(v,i) - r(v))) userDenominator = sum(r(u,i) - r(i)) otherUserDenominator = sum(r(v,i) - r(v)) where, r(u,i): rating of the movie i by the user u r(u): average rating of the user u
• Store the userids with maximum correlation in map function called output.
• Get predictions of each movie by a user giving some ratings and its neighborhood: r(u,i) = r(u) + sum(sim(u,v) * (r(v,i) - r(v))) / sum(abs(sim(u,v))) sim(u,v): similarity between u and v users where, r(u,i): rating of the movie i by the user u r(u): average rating of the user u
• Generate recommendations for the user based on the number of recommendations required.

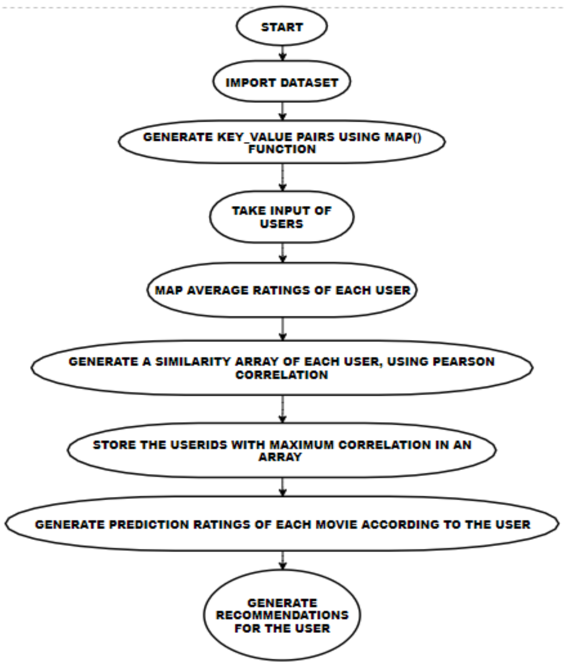

START
IMPORT DATASET
GENERATE KEY_VALUE PAIRS USING MAP() FUNCTION
TAKE INPUT OF USERS
MAP AVERAGE RATINGS OF EACH USER
GENERATE A SIMILARITY ARRAY OF EACH USER, USING PEARSON CORRELATION
STORE THE USERIDS WITH MAXIMUM CORRELATION IN AN ARRAY
GENERATE PREDICTION RATINGS OF EACH MOVIE ACCORDING TO THE USER
GENERATE RECOMMENDATIONS FOR THE USER

Fig. 3

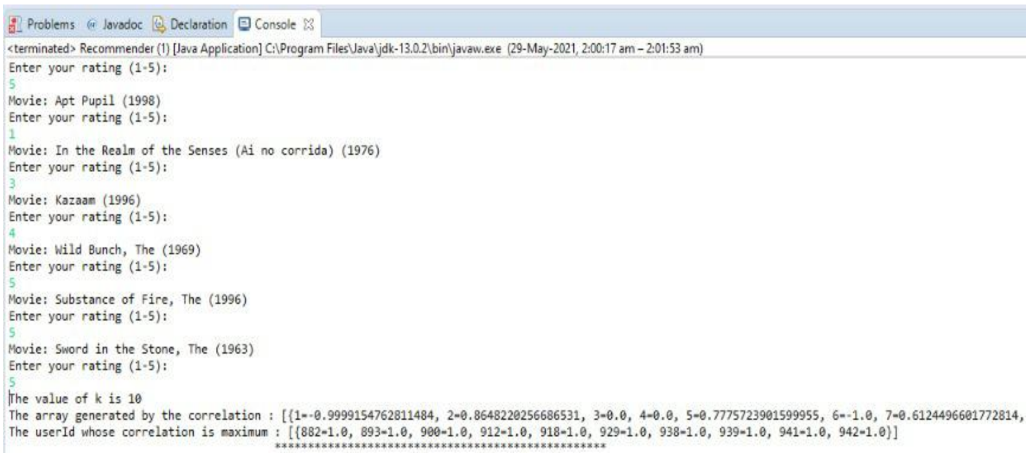
ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 11, November 2021

DOI: 10.17148/IJARCCE.2021.101127

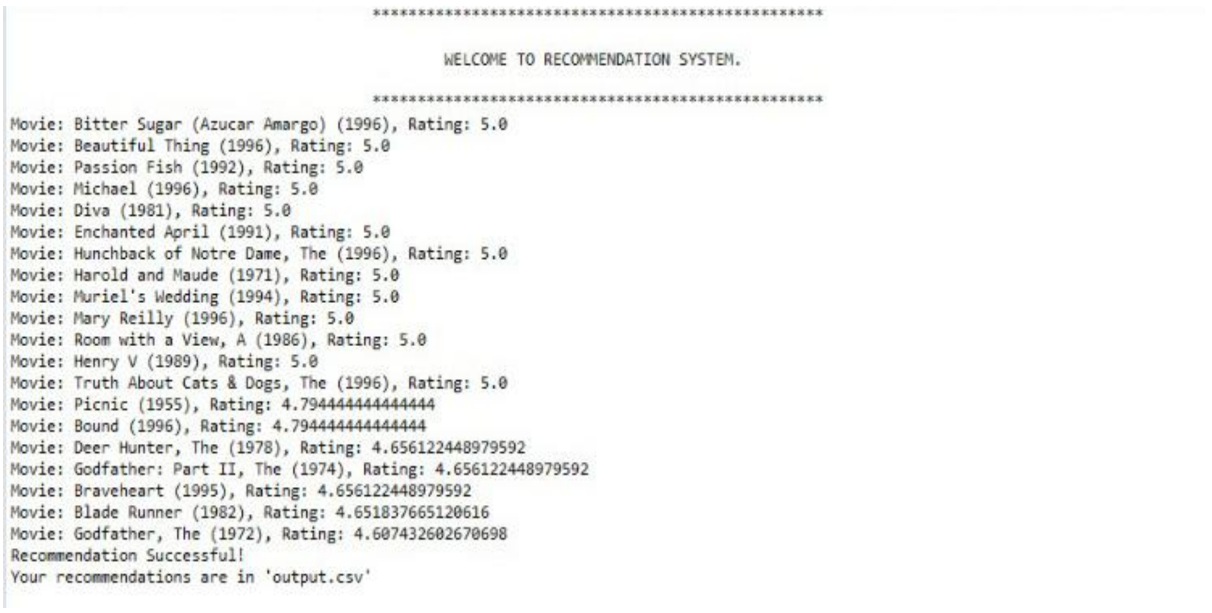
IV. RESULTS



```
<terminated> Recommender (1) [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (29-May-2021, 2:00:17 am – 2:01:53 am)
Enter your rating (1-5):
5
Movie: Apt Pupil (1998)
Enter your rating (1-5):
1
Movie: In the Realm of the Senses (Ai no corrida) (1976)
Enter your rating (1-5):
3
Movie: Kazaam (1996)
Enter your rating (1-5):
4
Movie: Wild Bunch, The (1969)
Enter your rating (1-5):
5
Movie: Substance of Fire, The (1996)
Enter your rating (1-5):
5
Movie: Sword in the Stone, The (1963)
Enter your rating (1-5):
5
The value of k is 10
The array generated by the correlation : [{1=-0.9999154762811484, 2=0.8648220256686531, 3=0.0, 4=0.0, 5=0.7775723901599955, 6=-1.0, 7=0.6124496601772814,
The userId whose correlation is maximum : [{882=1.0, 893=1.0, 900=1.0, 912=1.0, 918=1.0, 929=1.0, 938=1.0, 939=1.0, 941=1.0, 942=1.0}]
                                      **************************************************
```

Fig. 4

After taking ratings of certain movies from the user, k value is generated by the correlation and the user is recommended movies that are likely to match his/her interests.



```
                                        **************************************************

                                             WELCOME TO RECOMMENDATION SYSTEM.

                                        **************************************************
Movie: Bitter Sugar (Azucar Amargo) (1996), Rating: 5.0
Movie: Beautiful Thing (1996), Rating: 5.0
Movie: Passion Fish (1992), Rating: 5.0
Movie: Michael (1996), Rating: 5.0
Movie: Diva (1981), Rating: 5.0
Movie: Enchanted April (1991), Rating: 5.0
Movie: Hunchback of Notre Dame, The (1996), Rating: 5.0
Movie: Harold and Maude (1971), Rating: 5.0
Movie: Muriel's Wedding (1994), Rating: 5.0
Movie: Mary Reilly (1996), Rating: 5.0
Movie: Room with a View, A (1986), Rating: 5.0
Movie: Henry V (1989), Rating: 5.0
Movie: Truth About Cats & Dogs, The (1996), Rating: 5.0
Movie: Picnic (1955), Rating: 4.794444444444444
Movie: Bound (1996), Rating: 4.794444444444444
Movie: Deer Hunter, The (1978), Rating: 4.656122448979592
Movie: Godfather: Part II, The (1974), Rating: 4.656122448979592
Movie: Braveheart (1995), Rating: 4.656122448979592
Movie: Blade Runner (1982), Rating: 4.651837665120616
Movie: Godfather, The (1972), Rating: 4.607432602670698
Recommendation Successful!
Your recommendations are in 'output.csv'
```

Fig. 5

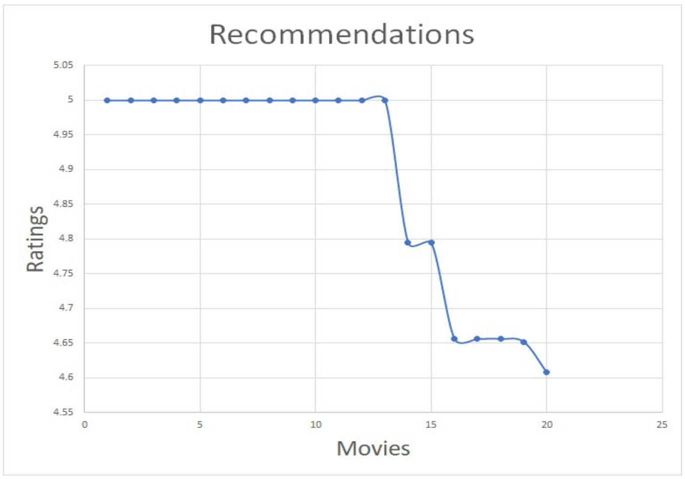


Fig. 6 The graph shows the recommended movies with their respective ratings.

ISSN (O) 2278-1021, ISSN (P) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 10, Issue 11, November 2021
DOI: 10.17148/IJARCCE.2021.101127

IV. CONCLUSION AND FUTURE SCOPE

• A recommendation system is a type of information filtering system which challenges to assume the priorities of a user, and make recommendations on the basis of user's priorities.
• The algorithm segregates the list of movies from the dataset according to the inputs provided by user and finally displays the list of movies.
• A bright feature of allowing the user to rate movies has enhanced the beauty of this recommender system.

REFERENCES

[1]. http://www.zheng-wen.com/WenRecommendation.pdf
[2]. http://ceur-ws.org/Vol-307/paper04.pdf
[3]. https://iopscience.iop.org/article/10.1088/1742-6596/1757/1/012168/pdf
[4]. Liwei H, Bitao J, Shouye L and Yanbo L 2018J. Review of research on deep learningbased recommendation systems(vol 41,Online Publication) 22 pp4-6
[5]. Tsang E P K Foundations of constraint satisfaction (DBLP), 1993 semi-fragile watermarking technique for image authentication (2002 6th Ai~r~uulCoference on Signal Prucessiirg ,vol2) pp 1592-1595
[6]. https://stackoverflow.com/questions/6268956/understanding-the-pearson-correlationcoefficient
[7]. https://ieeexplore.ieee.org/document/7270736
[8]. https://www.javatpoint.com/classification-algorithm-in-machine-learning
[9]. https://azati.ai/recommendation-systems-benefits-and-issues/
[10]. https://www.kaggle.com/rounakbanik/movie-recommender-systems