# MOVIE RECOMMENDATION SYSTEM

A PROJECT REPORT

*Submitted by*

P. ANTONY JOE NIRMAL – RA2111047010167

PRITHVI RAJA SHAH – RA2111047010168

ABHIGNA PENUBARTHI – RA2111047010169

MAYANK KAMBOJ – RA2111047010170

CH. SIDDHARTH – RA2111047010171

*Under the guidance of*

Dr. P. SRIDEVI PONMALAR

(Assistant Professor, Department of Computational Intelligence)

*in partial fulfillment for the award of the degree*

*of*

BACHELOR OF TECHNOLOGY

in

ARTIFICIAL INTELLIGENCE

of

FACULTY OF ENGINEERING AND TECHNOLOGY

S.R.M. Nagar, Kattankulathur, Kancheepuram District

# SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

## ACKNOWLEDGEMENTS

We would like to express our deepest gratitude to our guide, Dr. P. SRIDEVI PONMALAR, her valuable guidance, consistent encouragement, personal caring, timely help and providing our with an excellent atmosphere for doing the project. All through the work, in spite of her busy schedule, she has extended cheerful and cordial support to our for completing this project work.

ABSTRACT

In this hustling world, entertainment is a necessity for each one of us to refresh our mood and energy. Entertainment regains our confidence for work, and we can work more enthusiastically. For revitalizing ourselves, we can listen to our preferred music or can watch movies of our choice. For watching favorable movies online, we can utilize movie recommendation systems, which are more reliable, since searching of preferred movies will require more and more time which one cannot afford to waste. In this paper, to improve the quality of a movie recommendation system, a Hybrid approach by combining content based filtering and collaborative filtering, using Support Vector Machine as a classifier and genetic algorithm is presented in the proposed methodology and comparative results have been shown which depicts that the proposed approach shows an improvement in the accuracy, quality and scalability of the movie recommendation system than the pure approaches in three different datasets. Hybrid approach helps to get the advantages from both the approaches as well as tries to eliminate the drawbacks of both methods.

**INTRODUCTION**

A recommendation system or recommendation engine is a model used for information filtering where it tries to predict the preferences of a user and provide suggests based on these preferences. These systems have become increasingly popular nowadays and are widely used today in areas such as movies, music, books, videos, clothing, restaurants, food, places and other utilities. These systems collect information about a user's preferences and behavior, and then use this information to improve their suggestions in the future.

## 1.1 PROBLEM STATEMENT

A large number of companies are making use of recommendation systems to increase user interaction and enrich a user's shopping experience. Recommendation systems have several benefits, the most important being customer satisfaction and revenue. Movie Recommendation system is very powerful and important system. But, due to the problems associated with pure collaborative approach, movie recommendation systems also suffer with poor recommendation quality and scalability issues. The goal of the project is to recommend a movie to the user. Providing related content out of relevant and irrelevant collection of items to users of online service providers.

## 1.2 PROPOSED SOLUTION

Create a web-based system that allows consumers to register and reserve automobiles online while also allowing the firm to manage its car rental business efficiently. To make the process of renting an automobile easier for consumers.

This project covers a wide range of topics, from business concepts to computer science, and it necessitates the completion of numerous studies in order to meet the project’s objectives.

## 1.3 OBJECTIVE

## Improving the Accuracy of the recommendation system

## Improve the Quality of the movie Recommendation system

## Improving the Scalability. Enhancing the user experience.

## 1.4 METHODOLOGY

The hybrid approach proposed an integrative method by merging k-means clustering method and genetic algorithm based weighted similarity measure to construct a movie recommendation system. The proposed movie recommendation system gives finer similarity metrics and quality than the existing Movie recommendation system but the computation time which is taken by the proposed recommendation system is more than the existing recommendation system. This problem can be fixed by taking the clustered data points as an input dataset The proposed approach is for improving the scalability and quality of the movie recommendation system .We use a Hybrid approach , by unifying Content-Based Filtering and Collaborative Filtering, so that the approaches can be profited from each other. For computing similarity between the different movies in the given dataset efficiently and in least time and to reduce computation time of the movie recommender engine we used cosine similarity measure.

# Movie Recommendation System by K-Means Clustering AND K-Nearest Neighbor :

A recommendation system collect data about the user’s preferences either implicitly or explicitly on different items like movies. An implicit acquisition in the development of movie recommendation system uses the user’s behavior while watching the movies. On the other hand, a explicit acquisition in the development of movie recommendation system uses the user’s previous ratings or history. The other supporting technique that are used in the development of recommendation system is clustering. Clustering is a process to group a set of objects in such a way that objects in the same clusters are more similar to each other than to those in other clusters. K Means Clustering along with K-Nearest Neighbor is implemented on the movie lens dataset in order to obtain the best-optimized result. In existing technique, the data is scattered which results in a high number of clusters while in the proposed technique data is gathered and results in a low number of clusters. The process of recommendation of a movie is optimized in the proposed scheme. The proposed recommender system predicts the user’s preference of a movie on the basis of different parameters. The recommender system works on the concept that people are having common preference or choice. These users will influence on each other’s opinions. This process optimizes the process and having lower RMSE.

# FILTERATION SYSTEMS :

# CONTENT BASED FILTERING

This filtration strategy is based on the data provided about the items. The algorithm recommends products that are **similar** to the ones that a user has liked in the **past**. This similarity (generally cosine similarity) is computed from the data we have about the items as well as the user’s past preferences.

A picture containing chart

Description automatically generated

# COLLABORATIVE FILTERING

This filtration strategy is based on the combination of the user’s behavior and comparing and contrasting that with **other users’**behavior in the database. The history of **all users**plays an important role in this algorithm.

Diagram

Description automatically generated

There are two kinds of collaborative filtering systems; user-based recommender and item-based recommender.

1. **User-based filtering:**

User-based preferences are very common in the field of designing personalized systems. This approach is based on the user's likings. The process starts with users giving ratings (1-5) to some movies. These ratings can be implicit or explicit. Explicit ratings are when the user explicitly rates the item on some scale or indicates a thumbs-up/thumbs-down to the item. Often explicit ratings are hard to gather as not every user is much interested in providing feedbacks. In these scenarios, we gather implicit ratings based on their behaviour. For instance, if a user buys a product more than once, it indicates a positive preference. In context to movie systems, we can imply that if a user watches the entire movie, he/she has some likeability to it. Note that there are no clear rules in determining implicit ratings. Next, for each user, we first find some defined number of nearest neighbours. We calculate correlation between users' ratings using Pearson Correlation algorithm. The assumption that if two users' ratings are highly correlated, then these two users must enjoy similar items and products is used to recommend items to users.

1. **Item-based filtering:**

Unlike the user-based filtering method, item-based focuses on the similarity between the item’s users like instead of the users themselves. The most similar items are computed ahead of time. Then for recommendation, the items that are most similar to the target item are recommended to the user.

# **SYSTEM REQUIREMENTS SPECIFICATION**

This chapter involves both the hardware and software requirements needed for the project and detailed explanation of the specifications.

**1. Hardware Requirements**

• A PC with Windows/Linux OS

• Processor with 1.7-2.4gHz speed

• Minimum of 8gb RAM

• 2gb Graphic card

**2. Software Specification**

• Anaconda distribution package (Jupyter Notebook)

• Python libraries

1. **Software Requirements**

**Python Libraries:**

For the computation and analysis we need certain python libraries which are used to perform analytics. Packages such as SKlearn, Numpy, pandas, Matplotlib, Flask framework, etc are needed.

**SKlearn:**

It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

**NumPy:**

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python.

**Pandas:**

Pandas is one of the most widely used python libraries in data science. It provides high-performance, easy to use structures and data analysis tools. Unlike NumPy library which provides objects for multi-dimensional arrays, Pandas provides in-memory 2d table object called Data frame.

# **SYSTEM ANALYSIS AND DESIGN**

1. Diagram

   Description automatically generatedDiagram

   Description automatically generated**SYSTEM ARCHITECTURE**
2. **DATAFLOW**

Diagram

Description automatically generated

**Diagram

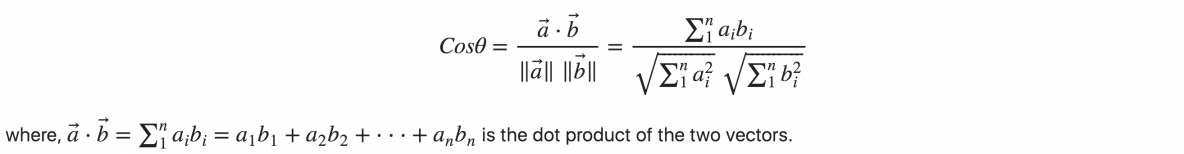
Description automatically generated**

**IMPLEMENTATION**

# **ALGORITHMS USED :**

**Cosine Similarity**:

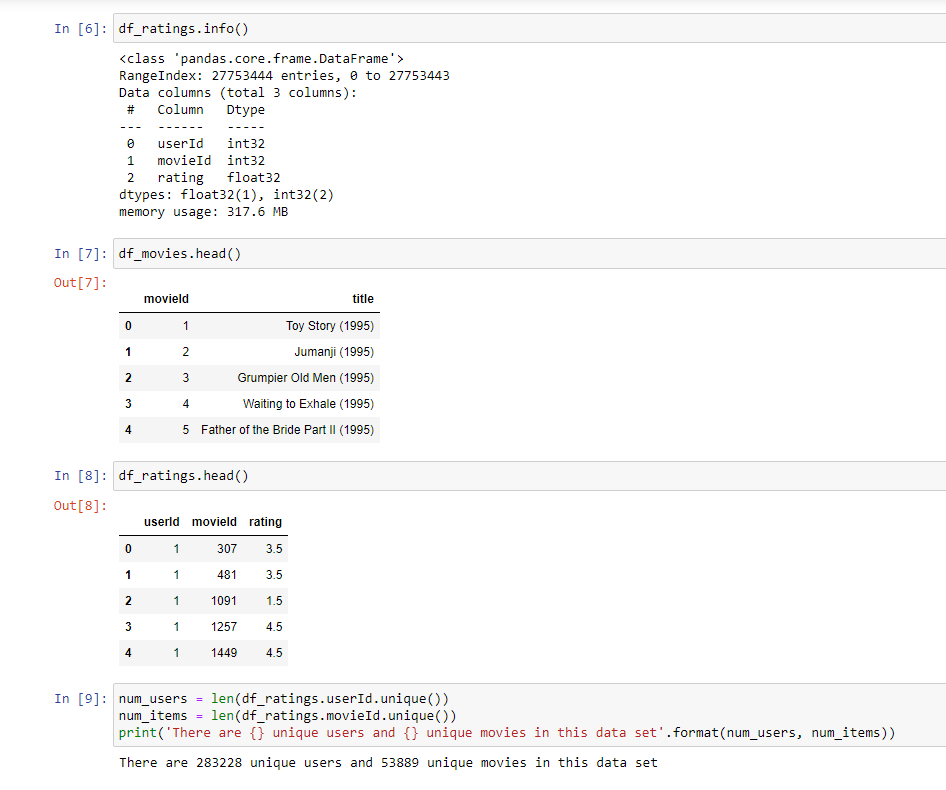
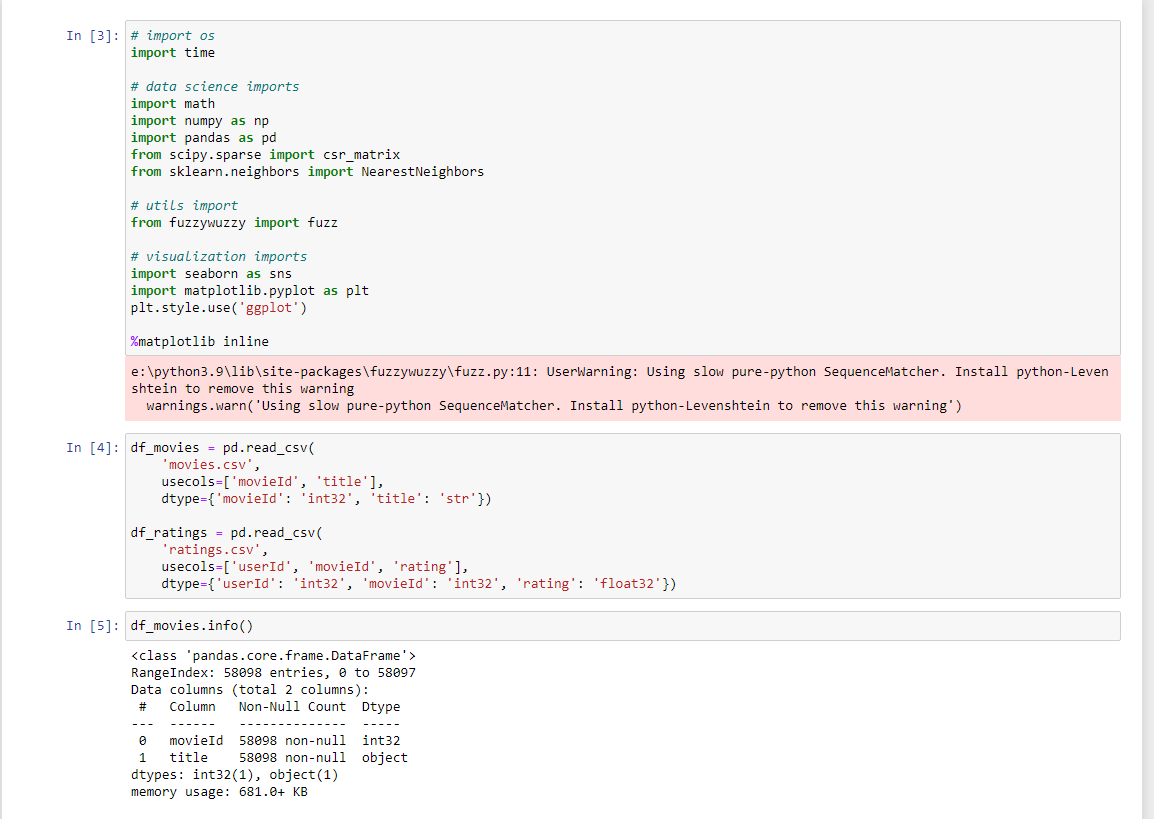
Cosine similarity is a measure of similarity between two non-zero vectors of an inner product space that measures the cosine of the angle between them.

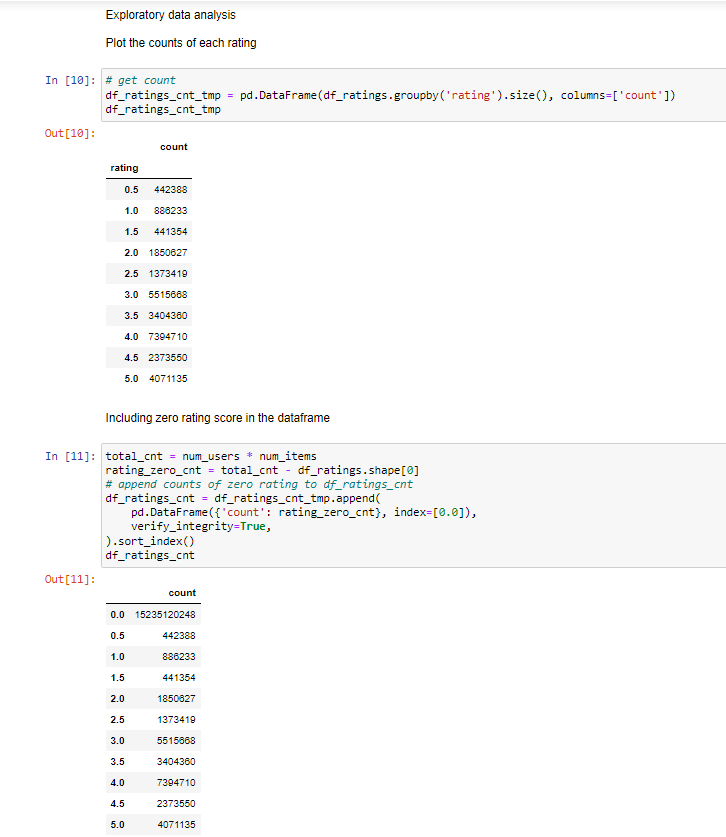
Formula:

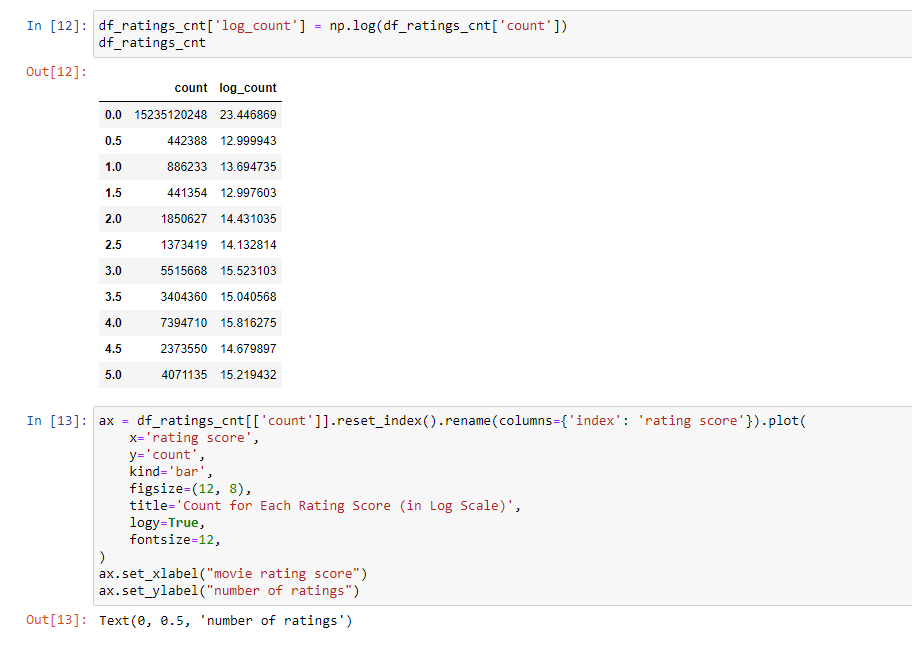
**Singular Value Decomposition (SVD):**

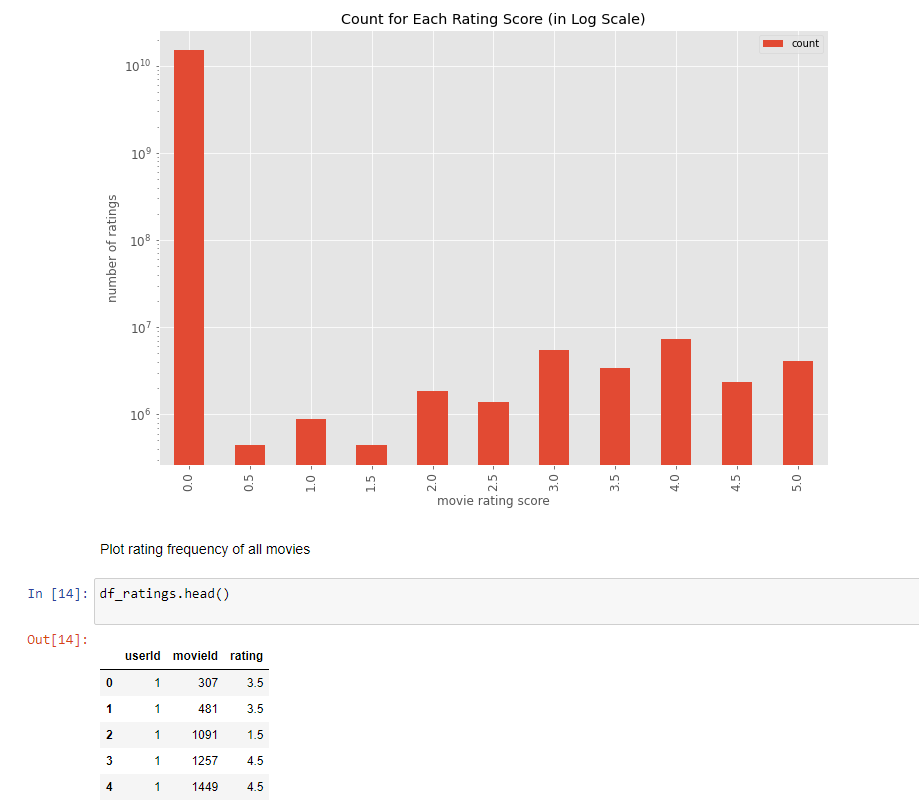
Let A be an n\*d matrix with singular vectors v1, v2, . . . , vr and corresponding singular values σ1, σ2, . . . , σr. Then ui = (1/σi )Avi , for i = 1, 2, . . . , r, are the left singular vectors.A picture containing text, watch

Description automatically generated

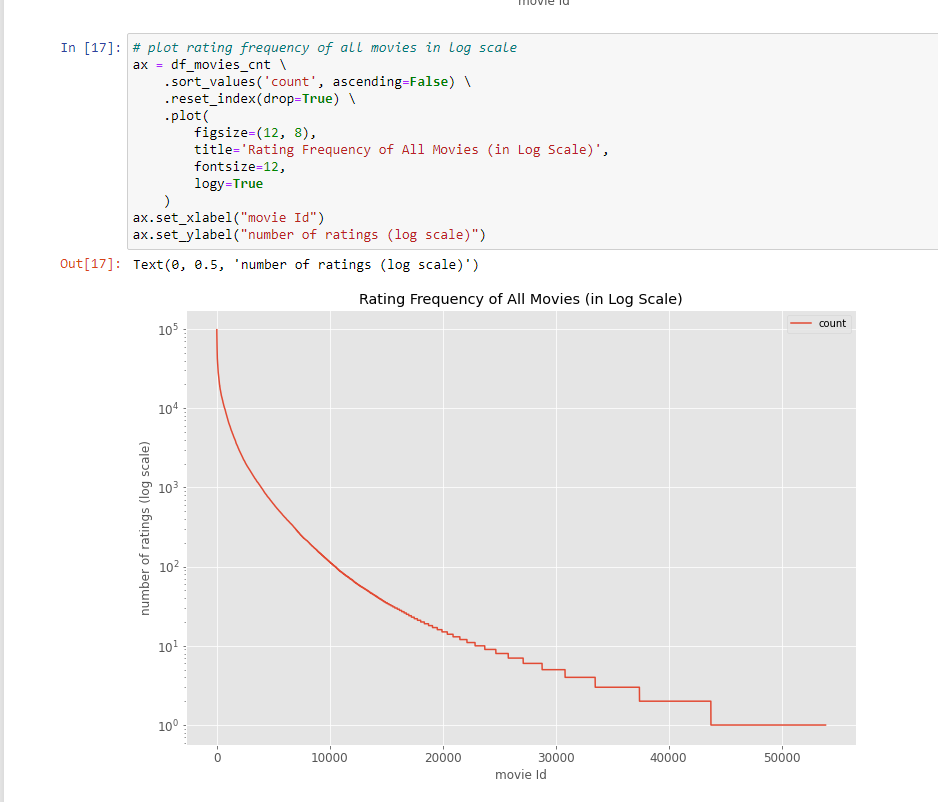








Graphical user interface, text, application, email

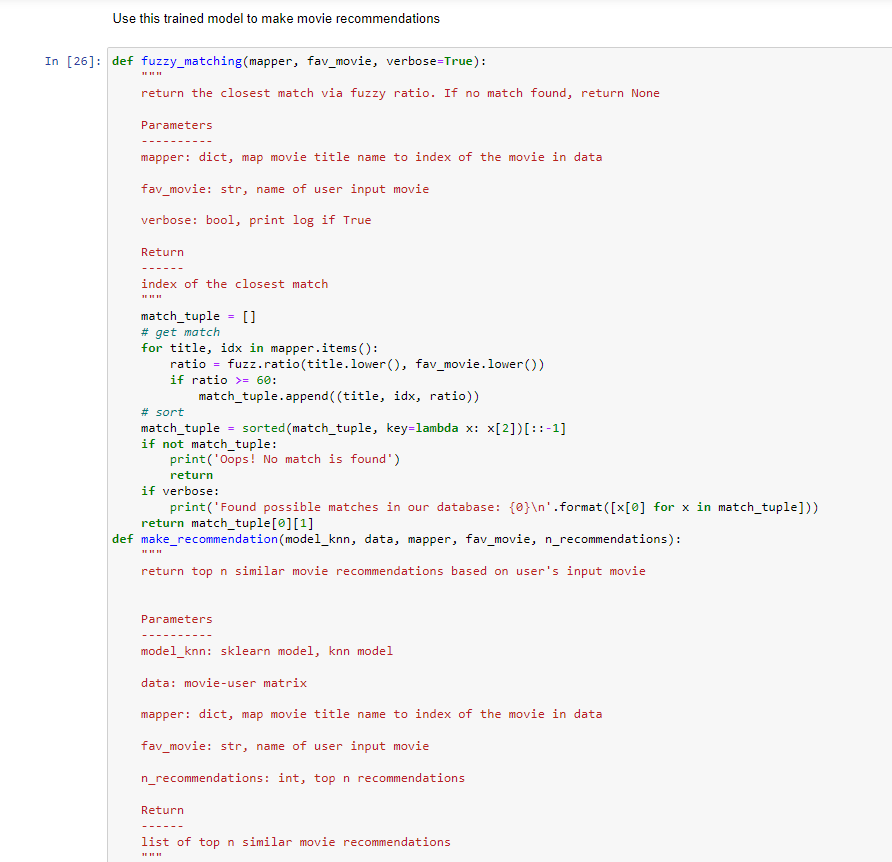
Description automatically generated

Graphical user interface, text, application

Description automatically generatedChart

Description automatically generated

Text

Description automatically generated with low confidence

Text

Description automatically generatedText

Description automatically generated

# **CONCLUSION**

In this project, to improve the accuracy, quality and scalability of movie recommendation system, a Hybrid approach by unifying content based filtering and collaborative filtering; using Singular Value Decomposition (SVD) as a classifier and Cosine Similarity is presented in the proposed methodology. Existing pure approaches and proposed hybrid approach is implemented on three different Movie datasets and the results are compared among them. Comparative results depicts that the proposed approach shows an improvement in the accuracy, quality and scalability of the movie recommendation system than the pure approaches. Also, computing time of the proposed approach is lesser than the other two pure approaches.