A Project Report

on

HYBRID MOVIE RECOMMENDATION SYSTEM

Submitted in partial fulfillment of the

requirement for the award of the degree of

Bachelor of Technology in Computer Science &

Engineering

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of
Dr.Amit Kumar Goel
Professor
Department of Computer Science and Engineering

Submitted By:

Abhinav Pengoria-18SCSE1010126

Abhishek Kumar-18SCSE1010334

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING /
DEPARTMENT OF COMPUTERAPPLICATION
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA

DECEMBER - 2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the project, entitled **"HYBRID MOVIE RECOMMENDATION SYSTEM"** in partial fulfillment of the requirements for the award of the **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING** submitted in the **School of Computing Science and Engineering of Galgotias University, Greater Noida,** is an original work carried out during the period of **SEP-2021 to DEC-2021,** under the supervision of **Dr.Amit Kumar Goel, Professor, Department of Computer Science and Engineering** of School of Computing Science and Engineering, Galgotias University, Greater Noida

The matter presented in the project has not been submitted by me/us for the award of any other degree of this or any other places.

Abhinav Pengoria - 18SCSE1010126
Abhishek Kumar - 18SCSE1010334

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Dr.Amit Kumar Goel
Professor

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of **Abhinav Pengoria- 18SCSE1010126, Abhishek Kumar - 18SCSE1010334** has been held on _________________ and his/her work is recommended for the award of **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING.**

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date: December, 2021

Place: Greater Noida

TABLE OF CONTENTS

ABSTRACT

These days, recommendation systems have made discovering the things simple that we need. Movie recommendation systems target helping movies devotees by recommending what movies to observe without having to go through the long process of choosing from a large set of movies that go up to thousands and millions that is time-consuming and confusing. In this project, our point is to lessen human exertion by proposing motion pictures dependent on the client's advantages. to handle such problems, we introduced a model combining both content-based and collaborative approaches. It will give logically express results contrasted with various frameworks that depend on a content-based approach. Content-based proposal frameworks are compelled to individuals, these frameworks don't endorse things out of the container, along these lines restricting your decision to investigate more. In all what we are trying to say is that we have focused on a framework that is able to settle these issues.

INTRODUCTION

A proposal framework is a sort of data sifting framework which endeavors to anticipate the inclinations of a client, and make recommends dependent on these inclinations. There are a wide assortment of utilization for suggestion frameworks. These have become progressively well known in the course of the most recent couple of years and are currently used in most internet based stages that we use. The substance of such stages differs from films, music, books and recordings, to companions and stories via online media stages, to items on web based business sites, to individuals on expert and dating sites, to indexed lists returned on Google. Frequently, these frameworks can gather data about a clients decisions, and can utilize this data to improve their ideas later on.For instance, Facebook can screen your cooperation with different stories on your feed to realize what sorts of stories appeal to you. Here and there, the recommender frameworks can make upgrades dependent on the exercises of an enormous number of individuals. For instance, if Amazon sees that countless clients who purchase the most recent Apple Macbook additionally purchase a USB-C-to USB Adapter, they can suggest the Adapter to another client who has recently added a Macbook to his truck. Because of the advances in recommender frameworks, clients continually anticipate great proposals. They have a low edge for administrations that can't make suitable ideas. On the off chance that a music streaming application isn't ready to anticipate and play music that the client likes, then, at that point, the client will basically quit utilizing it.This has prompted a high accentuation by tech organizations on further developing their proposal frameworks. Be that as it may, the issue is surprisingly complicated. Each client has various inclinations also, likes. Also, even the flavor of a solitary client can fluctuate contingent upon an enormous number of factors, like state of mind, season, or sort of action the client is doing. For instance, the sort of music one might want to hear while practicing varies significantly from the kind of music he'd pay attention to when preparing supper. Another issue that suggestion frameworks need to tackle is the investigation versus abuse issue.They should investigate new areas to find more concerning the client, while as yet capitalizing on what is as of now thought about of the client. Three fundamental methodologies are utilized for our recommender frameworks. One is Demographic Filtering i.e They submit summed up proposals to each client, in light of film ubiquity and additionally 7 type. The System prescribes similar motion pictures to clients with comparative segment highlights. Since every client is unique , this methodology is viewed as excessively straightforward. The fundamental thought behind this framework is that motion pictures that are more well known and widely praised will have a higher likelihood of being enjoyed by the normal crowd. Second is content-based sifting, where we attempt to profile the clients intrigues utilizing data gathered, and suggest things in view of that profile. The other is communicant sifting, where we attempt to bunch comparable clients together and use data about the gathering to make suggestions to the client.

1.1 Objectives

Hybrid Movies Recommendation System goal is to recommended movies by merging two recommendation system i.e, content based and collaborative filtering and create a system which possesses both system properties.

The main goal of the our system is:

I. To recommend movies.

II. To recommend movies based on user history i.e, the next movies recommended based on the number of times the user watches a certain genre of movie.

III. To recommend movies based on overall rating i.e, movies which are famous and these movies can have different genre also.

1.2 Problem Statement

Providing related content out of relevant and irrelevant collection of items to users of online service providers.

Hybrid movies recommendation system aims to recommend movies to users based on user-movie (item) ratings.

Given a set of users with their previous ratings for a set of movies, can we predict the rating they will assign to a movie they have not previously rated?

LITERATURE SURVEY

With regards to a survey of the writing, a suggestion framework utilizing a substance based collective and mixture approach by a past specialist is an alternate way to deal with the improvement of proposal based motors. In 2007 a web based and information based knowledge film suggestion framework has been offered utilizing the crossover sifting technique . In 2017, a film proposal framework upheld style and rating coefficient of connection reason by the creators. In 2013 a Bayesian organization and trust model based film suggestion motor have been prescribed to anticipate evaluations for clients and things, essentially from datasets to suggest clients their decision as well as the other way around.In 2018, the creators constructed a suggestion motor by investigating the evaluations data set gathered from Kaggle to suggest films for a client chosen from Python. In 2018 film suggestion motors give a cycle to assist clients classify clients with comparable k-mean cuckoo esteems and support learning based recommender frameworks, which are utilizing bicycling procedures . Starting exploration mostly focused on the substance of the suggestion framework that inspected the provisions of the item to finish the proposal task. Tests checked that their methodologies were more flexible and exact. Bayesian networks are utilized for model-put together inclinations based with respect to their specific situation.At the point when customers embrace new conduct, it is hard for community sifting to respond in a flash. Accordingly, the two analysts and professionals want to adjust community sifting strategy and content-based technique to address the issue. Ternary executed Unplugged Learning of Machine Learning to inspect the extremity of machine reflectivity.

2.1 Background

Over the past decade, a large number of recommendation systems for a variety of domains have been developed and are in use. These recommendation systems use a variety of methods such as content based approach, collaborative approach, knowledge based approach, utility based approach, hybrid approach, etc. Most of the online recommendation systems for a variety of items use ratings from previous users to make recommendations to current users with similar interests. One such system was designed by Jung, Harris, Webster and Her locker (2004) for improving search results. The system encourages users to enter longer and more informative search queries, and collects ratings from users as to whether search results meet their information need or not. These ratings are then used to make recommendations to later users with similar needs.

2.2 Existing product and system

Current Social Networking World Internet social networking sites, which began in 1995 with Classmates.com, have surged in popularity and use through word-of-mouth advertising. Since then, a wide range of virtual communities have formed serving different purposes and targeting varying niche audiences:recommendation system and haven't taken advantage of social-networking communities or crowd wisdom. Some websites, such as Blockbuster, do provide individualized recommendations based on a user's ratings but do not include any social networking component. Yahoo! Movies goes further and uses personal ratings to suggest movies currently playing in theatre, on TV, and out on DVD. It also draws upon its vast user base to give lists of similar movie fans, their ratings, and reviews. Other movie sites, like Flixster, take a different approach. Flixster forms web based communities around movies and suggests movies to watch based on what your friends have rated.

2.3 Movie Recommendation System by K-Means Clustering AND K-Nearest Neighbour.

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 behaviour 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 Neighbour 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.

2.4 Movie Recommendation System Using Collaborative Filtering:

Collaborative filtering systems analyse the user's behaviour and preferences and predict what they would like based on similarity with other users. There are two kinds of collaborative filtering systems; user-based recommender and item-based recommender.

A. Use-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 feedback's. 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 like ability 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 neighbors. 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.

B. 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.

2.5 System Study

Feasibility Study

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are:

Economical Feasibility

Technical Feasibility

Social Feasibility

Economical Feasibility:

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

Technical Feasibility:

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

Social Feasibility:

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

2.6 Hybrid Recommendation And Approach

One common occurrence in RSS research is the demand to combine recommendation techniques to achieve peak performance. All of the known recommendation techniques have advantages and disadvantages, and many researchers have chosen to combine techniques in different ways in order to leverage their advantages. This session surveys the different hybrid recommendation approaches.

Hybrid systems combine two or more techniques in order to gain better performance with fewer limitations of each approach. Many hybrid systems have been applied to travel and tourism applications. For instance F. Ricci et al. illustrate a travel planning.recommender system that is case-based, hence is knowledge-based, but also Collaborative-based since it recommends travel services that have been evaluated positively by others.

Fab is a recommendation system designed to help users explore the enormous amount of information available on the internet. This hybrid system combines the Content-based and Collaborative methods of recommendation in a way that exploits the advantages of the two approaches while avoiding their shortcomings. Fab's hybrid structure allows for automatic recognition of emergent issues relevant to various groups of users. It also enables two scaling problems pertaining to the rising number of users and documents, to be addressed.

One major tactic for improving recommendation is to combine Collaborative filtering with Content-based recommenders. We can illustrate the benefits of such hybrid systems with a simple example; suppose one user has rated the NBA page from CBSSports.com favorably, while another has rated the NBA page from CNNSL.com favorably, pure Collaborative filtering would find no correlation between the two users. However, Content analysis can show that the two items are in fact quite similar, thus indicating.

2.7 Survey of Technologies

Hybrid Movies Recommendation System goal is to recommended movies by merging two recommendation system i.e, content based and collaborative filtering and create a system which possesses both system properties.

PYTHON :

- Python is a popular programming language.
- Python is an interpreted high-level general-purpose programming language.
- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.
- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
- Python has a simple syntax similar to the English language.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.

- Python can be treated in a procedural way, an object-oriented way or a functional way.
- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.
- Python Comments can be used to explain Python code.
- Python Comments can be used to make the code more readable.
- Python Comments can be used to prevent execution when testing code.
- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- It supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.
- Python supports multiple programming pattern, including object-oriented, imperative, and functional or procedural programming styles.
- Python is not intended to work in a particular area, such as web programming. That is why it is known as multipurpose programming language because it can be used with web, enterprise, 3D CAD, etc.

REQUIREMENTS AND ANALYSIS

Problem Definition

This paper is based on recommendation system that recommends different things to users. This system will recommend movies to users. This system will provide more precise results as compared to the existing systems. The existing system works on individual users' rating. This may be sometime useless for the users who have different taste from the recommendations shown by the system as every user may have different tastes. This system calculates the similarities between different users and then recommend movie to them as per the ratings given by the different users of similar tastes. This will provide a precise recommendation to the user.

Providing related content out of relevant and irrelevant collection of items to users of online service providers.

Hybrid movies recommendation system aims to recommend movies to users based on user-movie (item) ratings.

Given a set of users with their previous ratings for a set of movies, can we predict the rating they will assign to a movie they have not previously rated?

Requirement Specification

Hybrid Movies Recommendation System provides the Customers with better and more efficient means of movies recommendation system and from all over the world. Technologies such as python.

- Python is a popular programming language.
- Python is an interpreted high-level general-purpose programming language.
- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.

SOFTWARE AND HARDWARE REQUIREMENTS

Hardware Used

1. Intel Core 2 Quad Q9550,E0 revision
2. 4 GB PC2-6400 800Mhz RAM
3. 500 GB Hard disk drive

Software Requirements

1. Windows 7 Ultimate
2. IE8/Firefox/Safari

Preliminary Product Description

We indulged ourselves into a lot of research before we started the actual work on the — "HYBRID MOVIES RECOMMENDATION SYSTEM" application. We Studied about the different kinds of movies recommendation system such as content based and collaborative filtering that were needed to build our application.

CONCEPTUAL MODELS

DATA FLOW DIAGRAM

1-Data Flow

An arrow represents data flow; it represents the path over which data travels in the system? A data flow can move between processes, flow into or out of data stores, to and from external entities.

2-Bubbles (Process)

A circle or bubble represents that transforms data from once form to another by performing some tasks with the data.

3-Data store

A data store is a place where data is held temporarily from one transaction to the next or is stored permanently.

4-Entity

External Entity symbol represents sources of data to the system or destinations of data from the system.

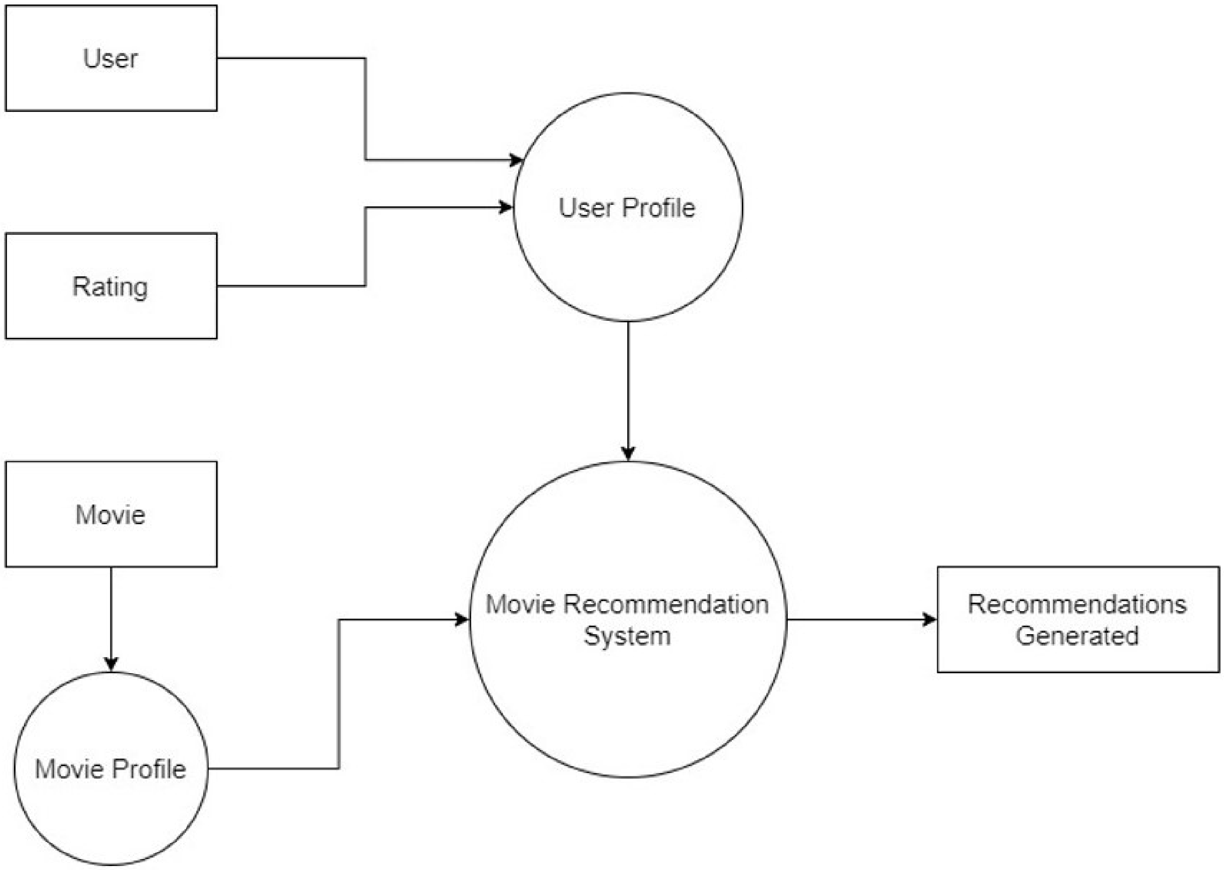


Fig:1 Data Flow Diagram

E-R DIAGRAM

1-Entity

Entity has a set of attribute whose value is uniquely, identify the entity or distinguish the entity from the other thing or entity in the world.

2-Attribute

Are used to define an entity that is the property that describe an entity. That is the properly that describe an entity.

3-Relationship

Is association among different/several,entities. It connect to one or more entities.

4-Derived

Attribute Any attribute which is derived from the relationship is called derived attribute.

5-Cardinality

Carnality refers to the multiplicity of the entities. How many entity are actually engaged in the relationship.

Types of cardinality:-

1. Many to 1
2. 1 to 1
3. 1 to Many
4. Many to Many

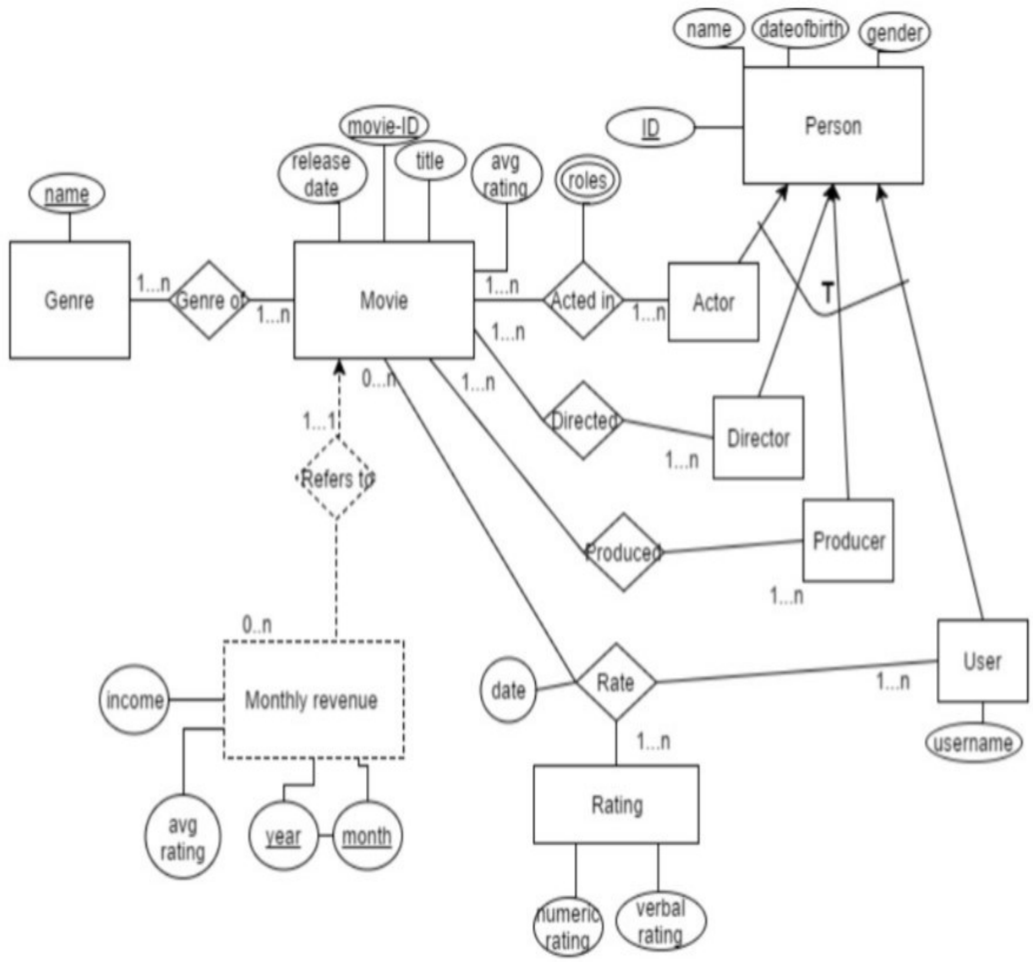


Fig:2 ER Diagram

CLASS DIAGRAM

Class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

- The above class diagram describes the classes, attributes and methods of movie recommendation system.

- Input movie data: It is class which takes the data of the movie details.

- A priori algorithm: This class will do analyses based on the given data and suggests movie.

- Movie selection: This class helps in selecting movie based on the genre and the movie ratings.

- Movie suggestion: This class suggests the movie to user based on the frequently watched movie(genre).

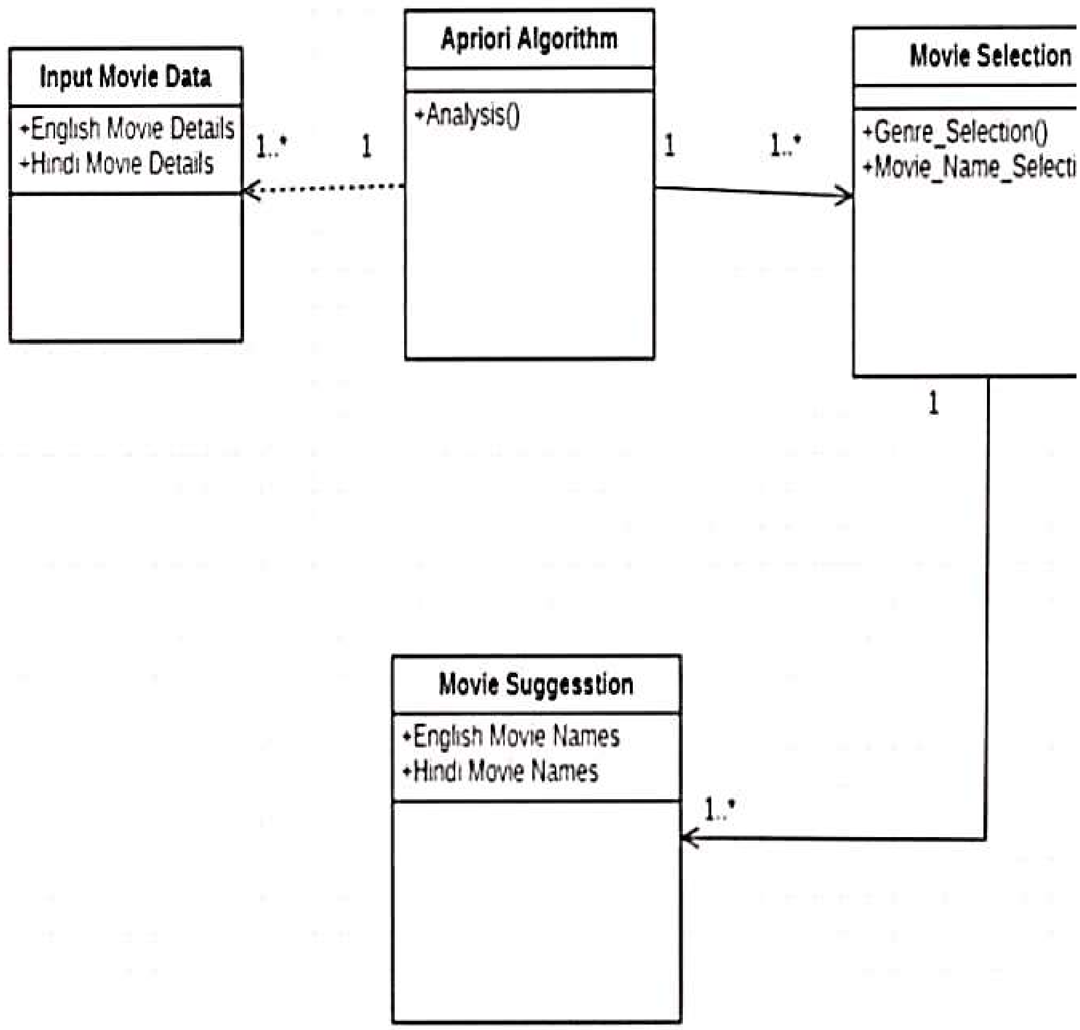


Fig:3 Class Diagram

STATE DIAGRAM

A state diagram is a diagram used in computer science to describe the behavior of a system considering all the possible states of an object when an event occurs. State diagrams graphically represent finite state machines. They are only used to understand object behavior throughout the whole system.

- In the above diagram it describes the behavior of movie recommendation system.
- When the data of movie is given it checks for the similar genre by using recommendation algorithm and suggests the movie to user.
- Suggestion is based on the movie ratings, genre, language and frequently watched movies.
- If the previously watched movie's rating is three, then the suggested movie's rating should be equal or greater than three.

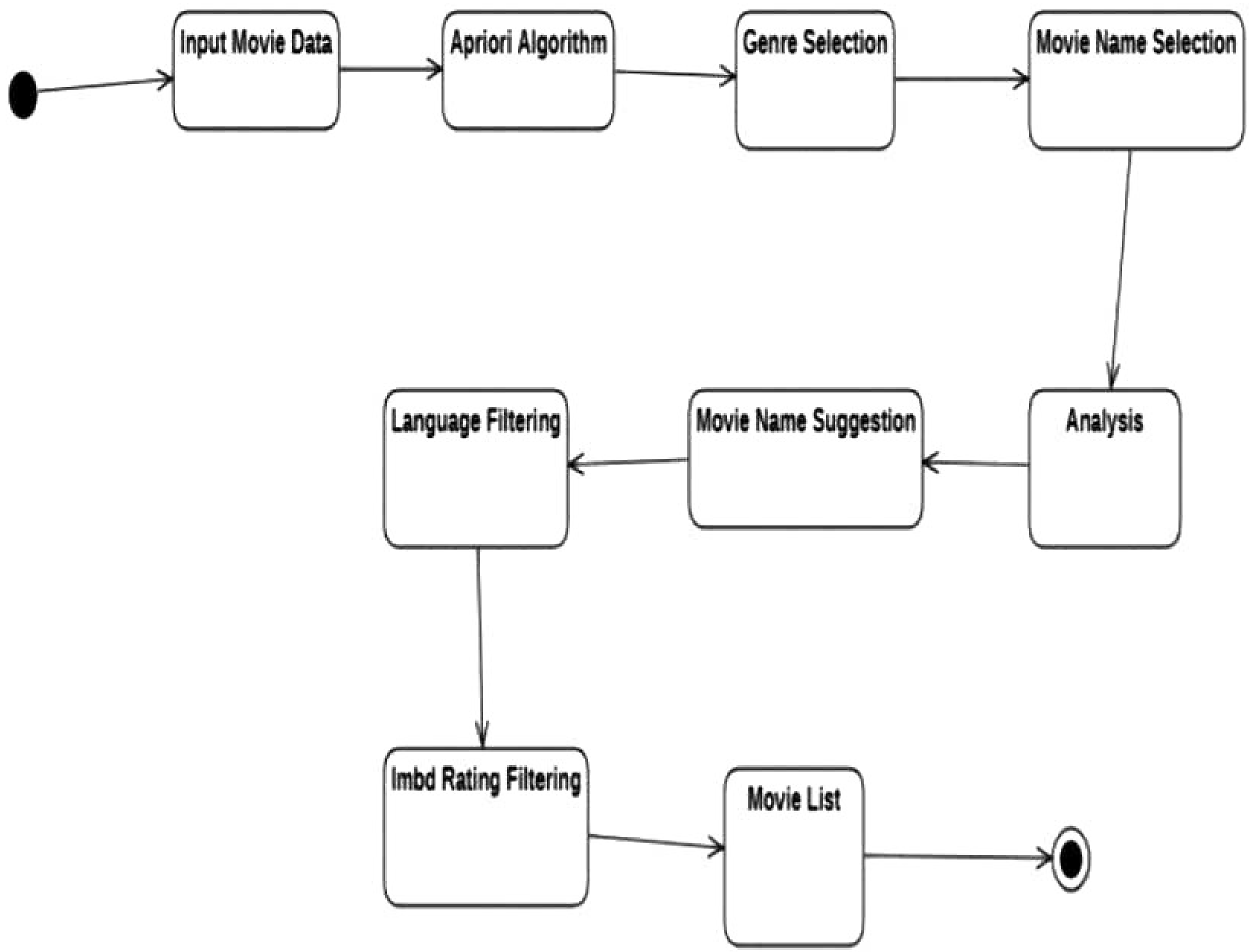


Fig:4 State Diagram

ARCHITECTURE DIAGRAM

An architecture diagram is a graphical representation of a set of concepts, that are part of an architecture, including their principles, elements and components. ... An Example Architecture Diagram of an Enterprise Architecture to create a Modern Smart and Green Company, using various concepts and principles.

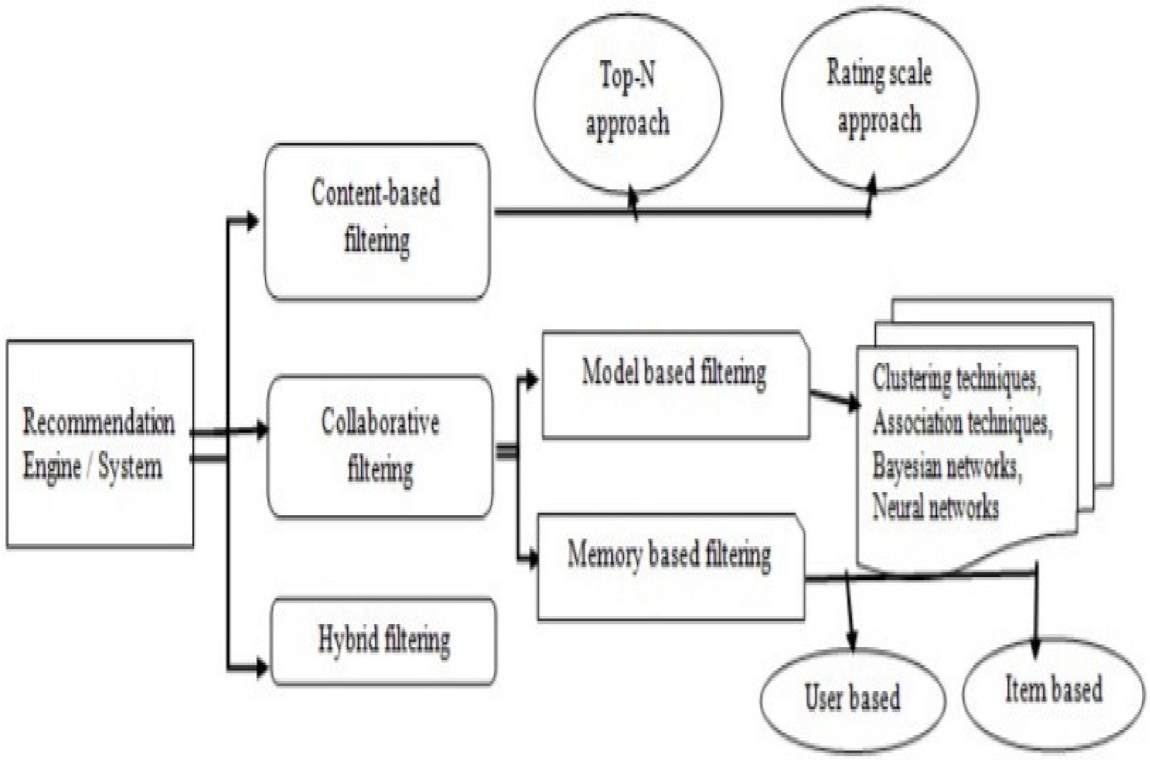


Fig:5 Architecture Diagram

SEQUENCE DIAGRAM

A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function.

- Once a movie is selected by the user the server gets the details of his previously watched movies.

- The information is stored and if we want to see the information again we can just check for the user history using the user id.

- For further the recommendation can be shared with friends on different platforms.

- The generation of recommendations goes on in loop until the user selects one for his watch.

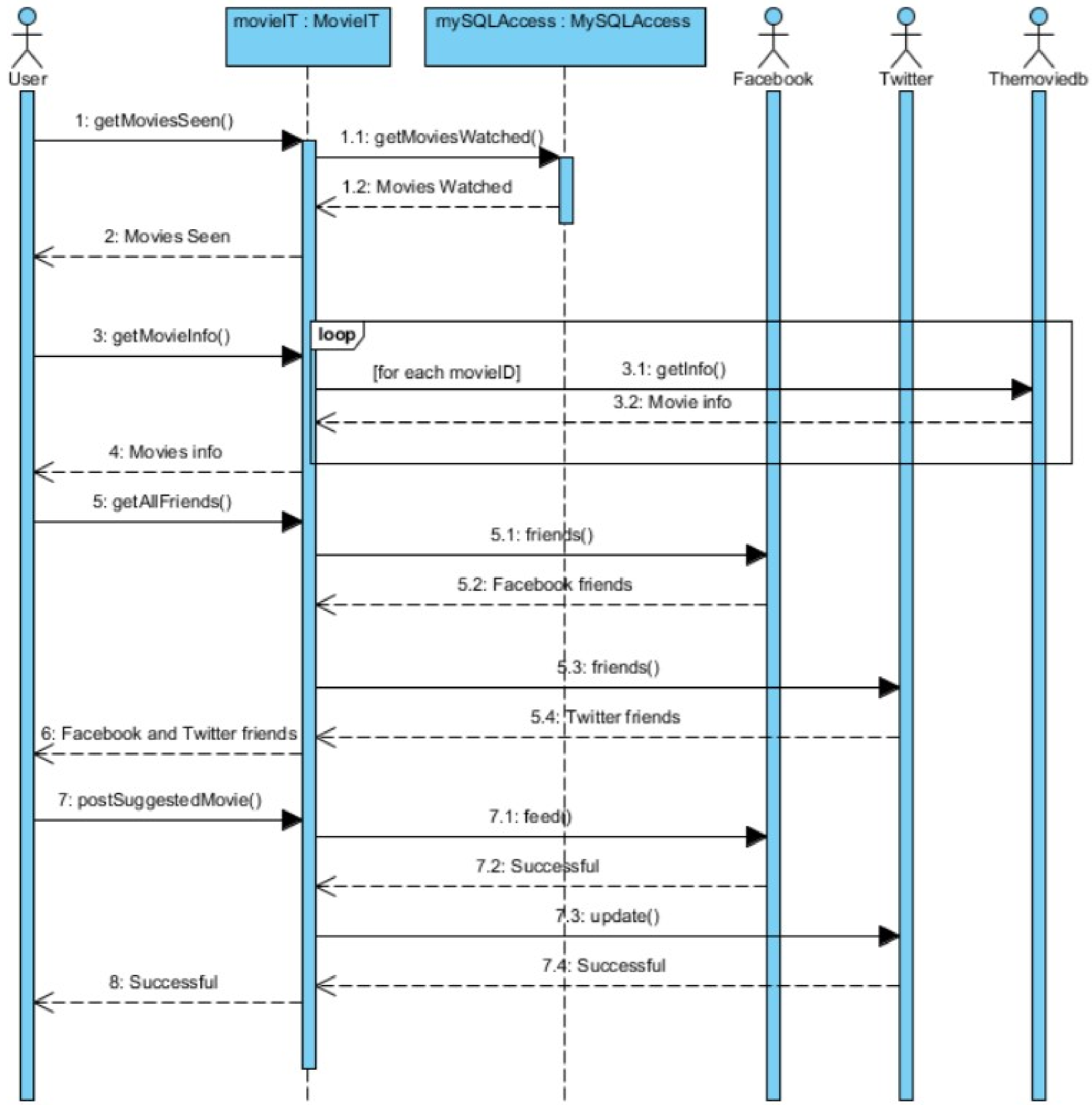


Fig:6 Sequence Diagram

USE CASE DIAGRAM

A use case shows a set of use cases, actors and their relationship. The use case diagram make system and classes approachable by presenting an outside view of how the elements may be used in context.

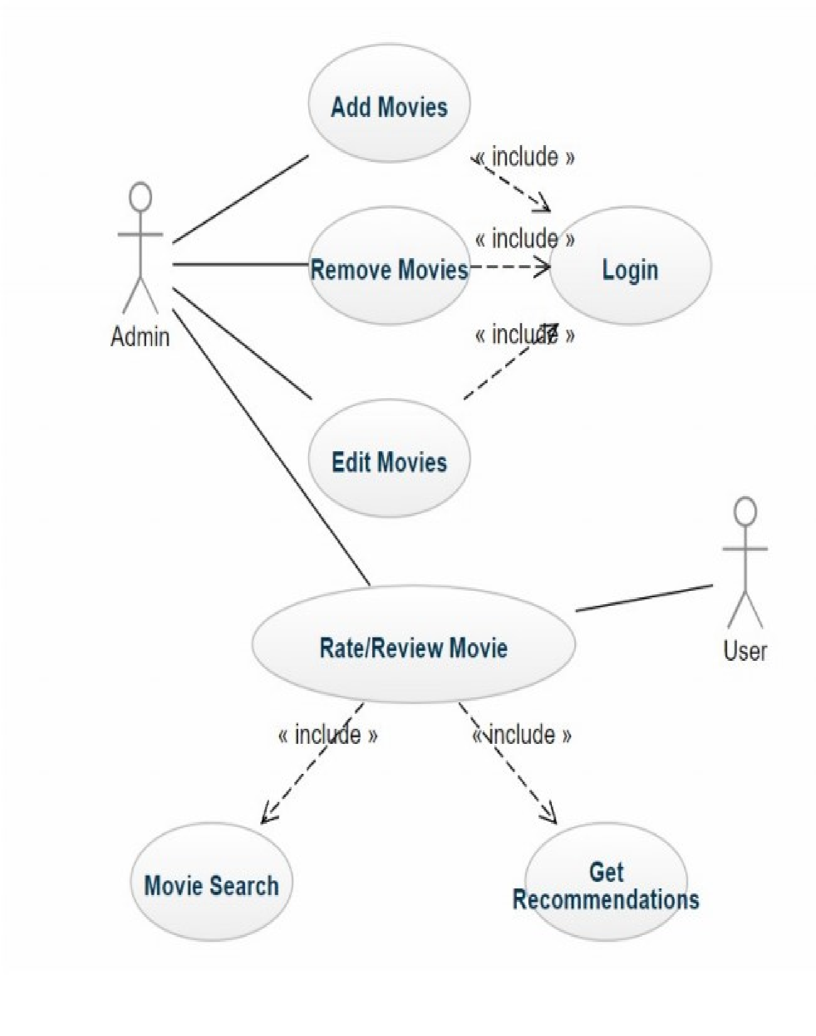


Fig:7 Use Case Diagram

FLOW CHART

A flowchart is a diagram that depicts a process, system or computer algorithm. They are widely used in multiple fields to document, study, plan, improve and communicate often complex processes in clear, easy-to-understand diagrams. Flowcharts, sometimes spelled as flow charts, use rectangles, ovals, diamonds and potentially numerous other shapes to define the type of step, along with connecting arrows to define flow and sequence. They can range from simple, hand-drawn charts to comprehensive computer-drawn diagrams depicting multiple steps and routes.

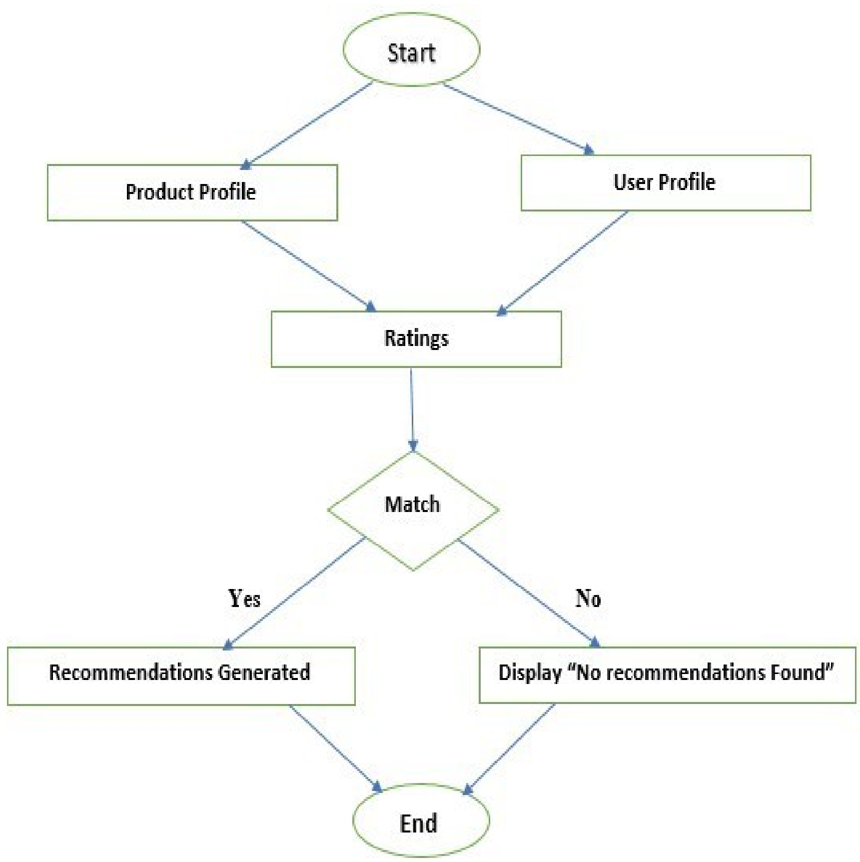


Fig:8 Flow Chart

ACTIVITY DIAGRAM

Activity diagram is defined as a UML diagram that focuses on the execution and flow of the behavior of a system instead of implementation. It is also called object-oriented flowchart. Activity diagrams consist of activities that are made up of actions which apply to behavioral modeling technology.

- Firstly, have to enter the user credentials.
- If a new user has to register.
- Then select the applicable genre and the movies are recommended.
- Identify candidate use cases, through the examination of business workflows.
- Identify pre- and post-conditions (the context) for use cases.
- Model workflows between/within use cases.
- Model complex workflows in operations on objects.
- Model in detail complex activities in a high level activity Diagram
- For this the system uses the A priori algorithm, and thus the process continues and always a new recommendation list is generated.

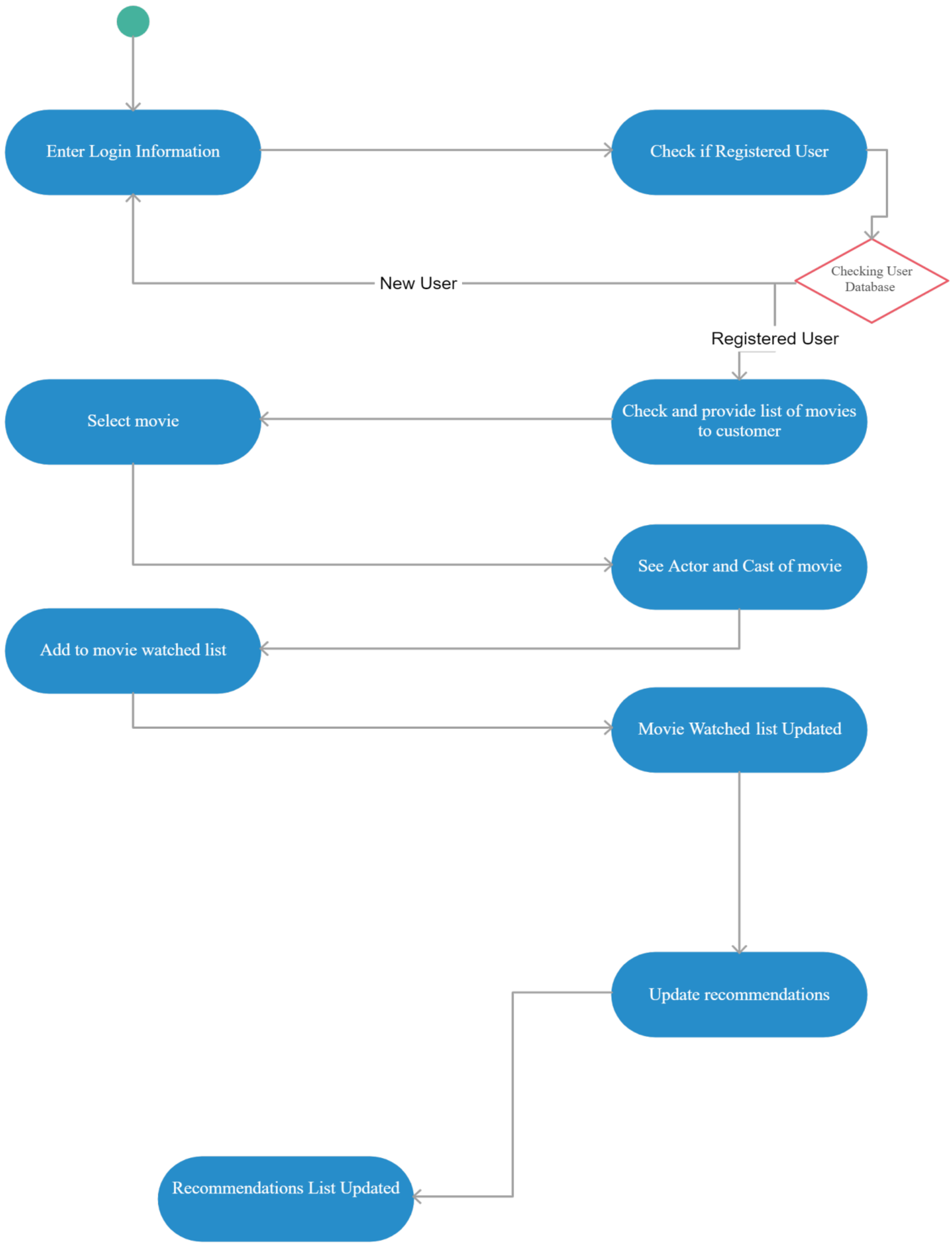
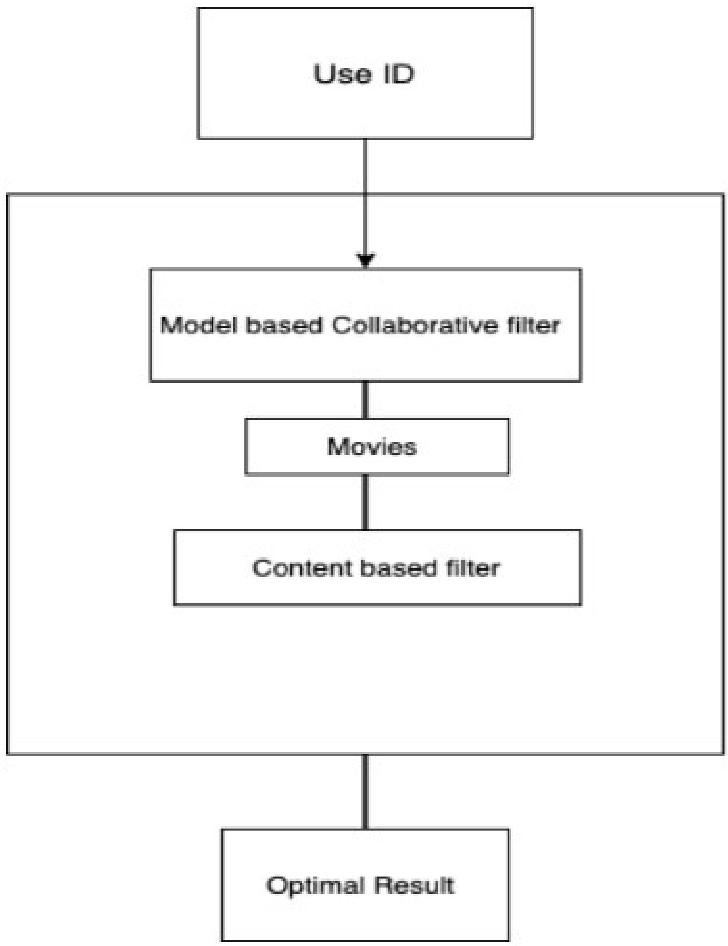


Fig:9 Activity Diagram

SYSTEM DESIGN

For each different individual use different list of movies are recommended ,as user login or enters the user id based on two different approaches used in the project each will recommend the set of movies to the particular user by combining the both the set of movie based on the user the hybrid model will recommend the single list of movie to the user.



IMPLEMENTATION AND TESTING

Python Code and Output:

```
import pandas as pd
import numpy as np
```

```
[ ] column_names = ['user_id', 'item_id', 'rating', 'timestamp']
    df = pd.read_csv('/content/u.data', sep='\t', names=column_names)
```

```
[1] df.head()
```

	user_id	item_id	rating	timestamp
0	0	50	5	881250949
1	0	172	5	881250949
2	0	133	1	881250949
3	196	242	3	881250949
4	186	302	3	891717742

```
movie_titles = pd.read_csv("/content/Movie_Id_Titles")
movie_titles.head()
```

	item_id	title
0	1	Toy Story (1995)
1	2	GoldenEye (1995)
2	3	Four Rooms (1995)
3	4	Get Shorty (1995)
4	5	Copycat (1995)

```
df = pd.merge(df,movie_titles,on='item_id')
df.head()
```

	user_id	item_id	rating	timestamp	title
0	0	50	5	881250949	Star Wars (1977)
1	290	50	5	880473582	Star Wars (1977)
2	79	50	4	891271545	Star Wars (1977)
3	2	50	5	888552084	Star Wars (1977)
4	8	50	5	879362124	Star Wars (1977)

```
import matplotlib.pyplot as plt
import seaborn as sns
sns.set_style('white')
%matplotlib inline
```

```
df.groupby('title')['rating'].mean().sort_values(ascending=False).head()
```

```
title
Marlene Dietrich: Shadow and Light (1996)     5.0
Prefontaine (1997)                            5.0
Santa with Muscles (1996)                     5.0
Star Kid (1997)                               5.0
Someone Else's America (1995)                 5.0
Name: rating, dtype: float64
```

```
df.groupby('title')['rating'].count().sort_values(ascending=False).head()
```

```
title
Star Wars (1977)             584
Contact (1997)               509
Fargo (1996)                 508
Return of the Jedi (1983)    507
Liar Liar (1997)             485
Name: rating, dtype: int64
```

```
ratings = pd.DataFrame(df.groupby('title')['rating'].mean())
ratings.head()
```

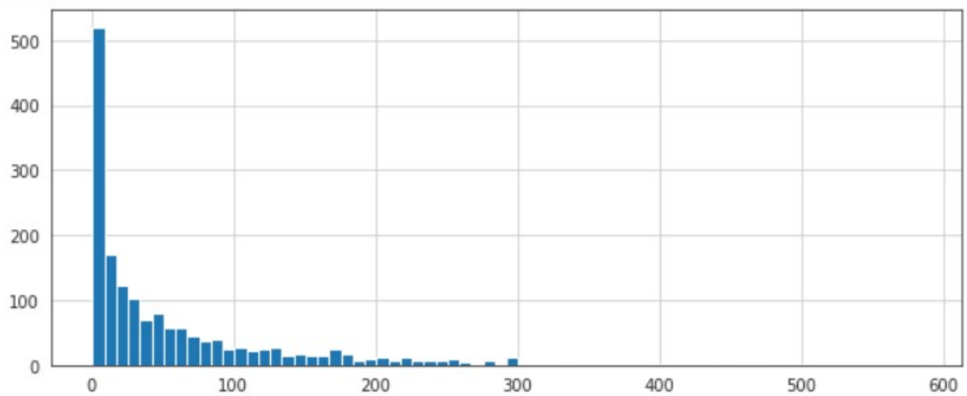
	rating
title	
'Til There Was You (1997)	2.333333
1-900 (1994)	2.600000
101 Dalmatians (1996)	2.908257
12 Angry Men (1957)	4.344000
187 (1997)	3.024390

```
[ ] ratings['num of ratings'] = pd.DataFrame(df.groupby('title')['rating'].count())
    ratings.head()
```

title	rating	num of ratings
'Til There Was You (1997)	2.333333	9
1-900 (1994)	2.600000	5
101 Dalmatians (1996)	2.908257	109
12 Angry Men (1957)	4.344000	125
187 (1997)	3.024390	41

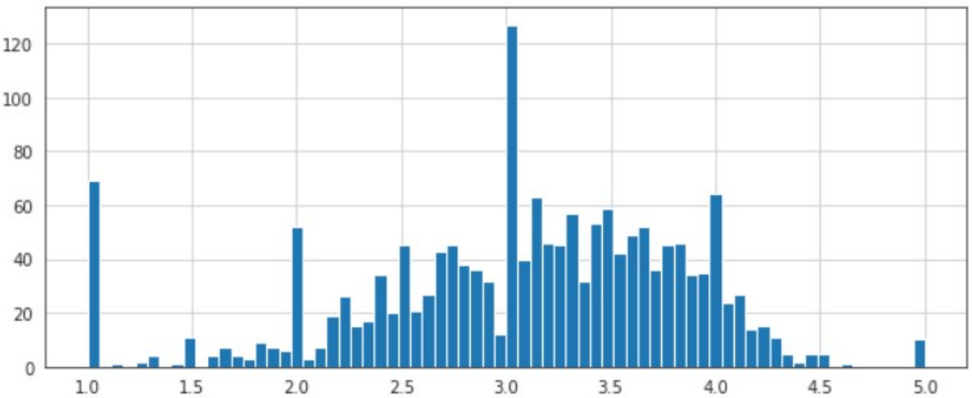
```
[ ] plt.figure(figsize=(10,4))
    ratings['num of ratings'].hist(bins=70)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fb81ff0fb10>
```



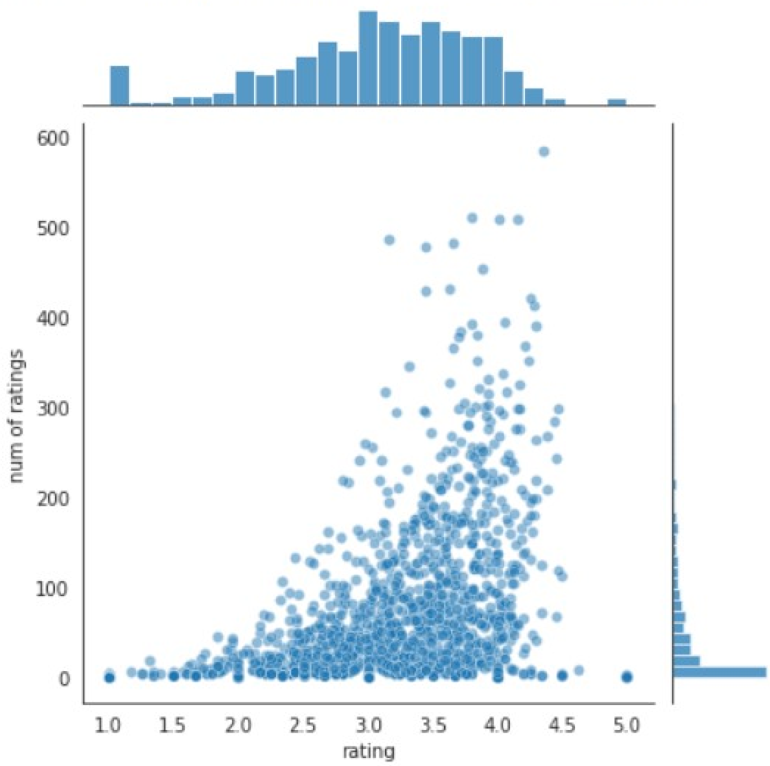
```
[ ] plt.figure(figsize=(10,4))
    ratings['rating'].hist(bins=70)
```

<matplotlib.axes._subplots.AxesSubplot at 0x7fb81feb8790>




```
[ ] sns.jointplot(x='rating',y='num of ratings',data=ratings,alpha=0.5)
```

<seaborn.axisgrid.JointGrid at 0x7fb81fd18a90>



```
moviemat = df.pivot_table(index='user_id',columns='title',values='rating')
moviemat.head()
```

title user_id	'Til There Was You (1997)	1-900 (1994)	101 Dalmatians (1996)	12 Angry Men (1957)	187 (1997)	2 Days in the Valley (1996)	20,000 Leagues Under the Sea (1954)	2001: A Space Odyssey (1968)	3 Ninjas: High Noon At Mega Mountain (1998)	39 Steps, The (1935)	8 1/2 (1963)	8 Heads in a Duffel Bag (1997)	8 Seconds (1994)	A Chef in Love (1996)	Above the Rim (1994)	Absolute Power (1997)	Abyss, The (1989)	Ace Ventura: Pet Detective (1994)	Ace Ventura: When Nature Calls (1995)
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	NaN	NaN	2.0	5.0	NaN	NaN	3.0	4.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	3.0	3.0	NaN
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	NaN	NaN	NaN	NaN	NaN	NaN	3.0	NaN	NaN	NaN
3	NaN	NaN	NaN	NaN	2.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

5 rows × 1664 columns

```
ratings.sort_values('num of ratings',ascending=False).head(10)
```

title	rating	num of ratings
Star Wars (1977)	4.359589	584
Contact (1997)	3.803536	509
Fargo (1996)	4.155512	508
Return of the Jedi (1983)	4.007890	507
Liar Liar (1997)	3.156701	485
English Patient, The (1996)	3.656965	481
Scream (1996)	3.441423	478
Toy Story (1995)	3.878319	452
Air Force One (1997)	3.631090	431
Independence Day (ID4) (1996)	3.438228	429

```
[ ] ratings.head()
```

title	rating	num of ratings
'Til There Was You (1997)	2.333333	9
1-900 (1994)	2.600000	5
101 Dalmatians (1996)	2.908257	109
12 Angry Men (1957)	4.344000	125
187 (1997)	3.024390	41

```
[ ] starwars_user_ratings = moviemat['Star Wars (1977)']
    liarliar_user_ratings = moviemat['Liar Liar (1997)']
    starwars_user_ratings.head()
```

```
user_id
0    5.0
1    5.0
2    5.0
3    NaN
4    5.0
Name: Star Wars (1977), dtype: float64
```

```
[19] corr_starwars = pd.DataFrame(similar_to_starwars,columns=['Correlation'])
     corr_starwars.dropna(inplace=True)
     corr_starwars.head()
```

title	Correlation
'Til There Was You (1997)	0.872872
1-900 (1994)	-0.645497
101 Dalmatians (1996)	0.211132
12 Angry Men (1957)	0.184289
187 (1997)	0.027398

```
[20] corr_starwars.sort_values('Correlation',ascending=False).head(10)
```

title	Correlation
Hollow Reed (1996)	1.0
Commandments (1997)	1.0
Cosi (1996)	1.0
No Escape (1994)	1.0
Stripes (1981)	1.0
Star Wars (1977)	1.0
Man of the Year (1995)	1.0
Beans of Egypt, Maine, The (1994)	1.0
Old Lady Who Walked in the Sea, The (Vieille qui marchait dans la mer, La) (1991)	1.0
Outlaw, The (1943)	1.0

```
[21] corr_starwars = corr_starwars.join(ratings['num of ratings'])
     corr_starwars.head()
```

title	Correlation	num of ratings
'Til There Was You (1997)	0.872872	9
1-900 (1994)	-0.645497	5
101 Dalmatians (1996)	0.211132	109
12 Angry Men (1957)	0.184289	125
187 (1997)	0.027398	41

```
[22] corr_starwars[corr_starwars['num of ratings']>100].sort_values('Correlation',ascending=False).head()
```

title	Correlation	num of ratings
Star Wars (1977)	1.000000	584
Empire Strikes Back, The (1980)	0.748353	368
Return of the Jedi (1983)	0.672556	507
Raiders of the Lost Ark (1981)	0.536117	420
Austin Powers: International Man of Mystery (1997)	0.377433	130

```
[23] corr_liarliar = pd.DataFrame(similar_to_liarliar,columns=['Correlation'])
     corr_liarliar.dropna(inplace=True)
     corr_liarliar = corr_liarliar.join(ratings['num of ratings'])
     corr_liarliar[corr_liarliar['num of ratings']>100].sort_values('Correlation',ascending=False).head()
```

title	Correlation	num of ratings
Liar Liar (1997)	1.000000	485
Batman Forever (1995)	0.516968	114
Mask, The (1994)	0.484650	129
Down Periscope (1996)	0.472681	101
Con Air (1997)	0.469828	137

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.

Future Scope of the Project

In the proposed approach, It has considered Genres of movies but, in future we can also consider age of user as according to the age movie preferences also changes, like for example, during our childhood we like animated movies more as compared to other movies. There is a need to work on the memory requirements of the proposed approach in the future. The proposed approach has been implemented here on different movie datasets only. It can also be implemented on the Film Affinity and Netflix datasets and the performance can be computed in the future.

REFERENCES

[1] Hirdesh Shivhare, Anshul Gupta and Shalki Sharma (2015), "Recommender system using fuzzy c-means clustering and genetic algorithm based weighted similarity measure", IEEE International Conference on Computer, Communication and Control.

[2] Manoj Kumar, D.K. Yadav, Ankur Singh and Vijay Kr. Gupta (2015), "A Movie Recommender System: MOVREC", International Journal of Computer Applications (0975 – 8887) Volume 124 – No.3.

[3] RyuRi Kim, Ye Jeong Kwak, Hyeon Jeong Mo, Mucheol Kim, Seungmin Rho,Ka Lok Man, Woon Kian Chong (2015),"Trustworthy Movie Recommender System with Correct Assessment and Emotion Evaluation", Proceedings of the International Multi Conference of Engineers and Computer Scientists Vol II.

[4] Zan Wang, Xue Yu*, Nan Feng, Zhenhua Wang (2014), "An Improved Collaborative Movie Recommendation System using Computational Intelligence",Journal of Visual Languages & Computing,Volume 25, Issue 6.

[5] Debadrita Roy, Arnab Kundu, (2013), "Design of Movie Recommendation System by Means of Collaborative Filtering", International Journal of Emerging Technology and Advanced Engineering, Volume 3, Issue 4.

[6] A. Javari, J. Gharibshah, and M. Jalili, "Recommender systems based on collaborative filtering and resource allocation," Social Network Analysis and Mining, Vol. 4, No. 1, pp. 234, 2014.

[7] M. Roth, A. Ben-David, D. Deutscher, G. Flysher, I. Horn, A. Leichtberg, and R. Merom, "Suggesting friends using the implicit social graph," in Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM: Washington, DC, USA., pp. 233-242, 2010.

[8] M. Li, BM. Dias, I. Jarman, W. El-Deredy, and P.J. Lisboa, "Grocery shopping recommendations based on basketsensitive random walk," in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM: Paris, France, pp. 1215-1224, 2009.

[9] H. Ma, and H. Liu, "Design of Clothing Clustering Recommendation System on SOM Neural Network," In 8th International Conference on Social Network, Communication and Education, Atlantis Press, 2018.

[10] H. Ma, H. Yang, M.R. Lyu, and I. King, "SoRec: social recommendation using probabilistic matrix factorization," in Proceedings of the 17th ACM conference on Information.